Citrix

MULTIUSER™

COMMAND REFERENCE

CITRIX

Citrix *MULTIUSER*

COMMAND REFERENCE

NOTICE

Citrix Systems, Inc., ("CITRIX"), makes no representations or warranties with respect to the contents or use of this publication, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, CITRIX reserves the right to make changes in specifications and other information contained in this publication without prior notice, and without obligation to notify any person or entity of such revisions or changes.

Patent #4,825,358
Patent #4,779,187

Citrix Systems, Inc.
210 University Drive, Suite 700
Coral Springs, FL 33071

Printed in the United States of America.

TRADEMARKS

The following names, used in this publication, are trademarks or registered trademarks:

Citrix and Citrix *MULTIUSER*	Citrix Systems, Inc.
Microsoft	Microsoft Corporation
MS	Microsoft Corporation
MS DOS	Microsoft Corporation
Codeview	Microsoft Corporation
MS OS/2	Microsoft Corporation
Microsoft Excel	Microsoft Corporation
Microsoft Word	Microsoft Corporation
IBM, PS/2, and OS/2	International Business Machines Corporation
IBM Quietwriter III	International Business Machines Corporation
PC/AT	International Business Machines Corporation
Micro Channel	International Business Machines Corporation
Presentation Manager	International Business Machines Corporation
Compaq	Compaq Computer Corporation
Hayes	Hayes Microcomputer Products Incorporated
386	Intel Corporation
National Semiconductor 16450	National Semiconductor Corporation
Esprit Opus 3n1+	Esprit Systems Incorporated
Kimtron KT-70PC	Kimtron Corporation
MC2, LINK MC2PC, and LINK MC5	Link Technologies Incorporated
IBM 3151	International Business Machines Corporation
ADDS 2025	Applied Digital Data Systems Incorporated
Wyse	Wyse Technology
Wyse 60 and Wyse 150	Wyse Technology
WY-60 and WY-150	Wyse Technology
Okidata and Okitel	Oki Electric Industry Company
Okidata Okitel 2400	Oki Electric Industry Company
Multitech and Multitech 224E	Multi-Tech Systems Incorporated
Televideo	Televideo Systems Incorporated
Televideo 965	Televideo Systems Incorporated
Boca Research IO/AT	Boca Research Incorporated
Digiboard and Digiboard PC/Xi	Digiboard Incorporated
Digiboard MC/Xi	Digiboard Incorporated
AST and AST I/O Mini II	AST Research Incorporated

TABLE OF CONTENTS

CHAPTER QUICK REFERENCE

WELCOME TO Citrix *MULTIUSER*

This reference describes Citrix *MULTIUSER* commands categorized into the following sections:

- Commands and utilities
- Configuration (*CONFIG.SYS*) commands
- Device drivers

The commands are in alphabetical order within each section. Chapter 1 describes each of the following categories.

- Commands
- Batch commands
- Utilities
- Multiuser utilities
- Configuration (*CONFIG.SYS*) commands
- Device drivers

This manual is intended as a reference for experienced users. For additional information about Citrix *MULTIUSER*, see the following manuals:

- To learn basic Citrix *MULTIUSER* skills and how to use Citrix *MULTIUSER*, see the *Citrix MULTIUSER User's Guide.*

- For a comprehensive description of Citrix *MULTIUSER* system installation, configuration, maintenance, and operation, see the *Citrix MULTIUSER System Administrator's Guide.*

Because Citrix *MULTIUSER* provides multiuser extensions to MS OS/2, it will be referred to in this publication as MS OS/2 *MULTIUSER.*

NOTATIONAL CONVENTIONS

To help you locate and interpret information easily, this reference uses specific typographic conventions and a standard syntax format and terminology. The following typographic conventions are used in this reference:

Text Element	Notational Convention
KEYS	Keys appear in boldface and uppercase.
variables	Variables are in lowercase italics.
USER INPUT	User input appears in uppercase and in a different typeface.
FILENAMES, PROGRAM NAMES, and *DEVICE DRIVERS*	Names of files, programs (including applications), and device drivers are in uppercase italics.
COMMANDS, DIRECTORY NAMES, DRIVE NAMES, and UTILITIES	These are always in uppercase.

Key Combinations

Key combinations and key sequences appear in the following format:

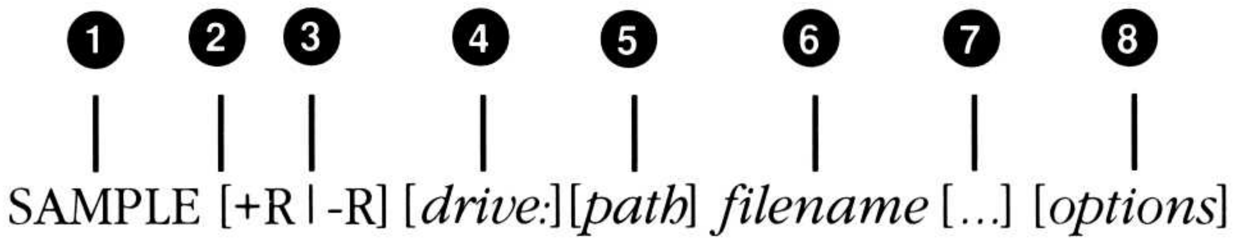
Notation	Meaning
KEY+KEY	A plus sign (+) between keynames means you must press the keys at the same time. For example, "Press **ALT+ESC**" means that you press the **ALT** key and hold it down while you press the **ESC** key.
KEY,KEY	A comma (,) between keynames means you must press the keys in sequence. For example, "Press **ALT,SPACEBAR**" means that you press the **ALT** key and release it, and then press the **SPACEBAR** and release it.
DIRECTION keys	Arrow keys on your computer keypad indicate DIRECTION. The name refers to the direction in which the arrow on the key points: **UP**, **DOWN**, **RIGHT**, or **LEFT**.

SYNTAX CONVENTIONS

Syntax represents the order in which you must type a command-line command or utility name and any arguments and options that follow it. Unless otherwise specified, you can type commands, arguments, and options in either uppercase or lowercase letters.

All commands, utilities, and configuration commands should be typed in on a single line on your terminal before pressing the **ENTER** key. If they do not fit on a single line, they can span multiple lines but the **ENTER** key should be pressed only at the end. In some cases in this book, the syntax spans multiple lines because it does not physically fit on a single line.

The following is a sample syntax line:



SAMPLE [+R | -R] [*drive:*][*path*] *filename* [...] [*options*]

The meaning of each of these elements is as follows:

variables — Elements in lowercase italics are variables for which you must supply the text. For example, when *filename* appears, you should type the name of your file.

Number	Element	Meaning
1	SAMPLE	Specifies the name of the command or utility. Elements shown in uppercase letters indicate the exact text to be entered. You may make your entry, however, either in uppercase or lowercase letters.
2	[ ]	Items in brackets are optional. To include the optional information described within the brackets, type only the information. Do not type the brackets themselves.

Number	Element	Meaning
3	|	A vertical bar means that you must choose from the option on either side of it. For example, ON | OFF means that you are to enter either ON or OFF. Do not type the pipe symbol itself.
4	*drive*:	Specifies a disk drive. You need to specify a drive name along with a *filename* only if you are using a file that is **not** on the current drive. The colon (:) must be typed as shown.
5	*path*	Specifies a complete directory path, using the following syntax: [\DIRECTORY...][\DIRECTORY...]\DIRECTORY You need to specify a path along with a *filename* only if the file is not in the current directory.

Number	Element	Meaning
6	*filename*	Specifies a file. In the file allocation table (FAT) file system, a *filename* can be up to eight characters long, followed by a period (.) and an extension of up to three characters (for example, *YOURFILE.EXT*). In the High-Performance File System (HPFS), a *filename* can be up to 254 characters long and can include periods and spaces in more than one place (for example, the name *YOUR.FILE WITH A LONG.FILENAME*), so there is no distinction in HPFS between a *filename* and an extension. An HPFS *filename* can also include certain characters not allowed in the FAT file system. The *filename* argument cannot be a device name or a drive letter.
7	...	An ellipsis indicates that an argument can be repeated as many times as necessary in a command line. Type only the information, not the ellipsis (...) itself.

Number	Element	Meaning
8	*options*	Specifies one or more command options. An option begins with a slash—for example, /P.

Other placeholders used in syntax lines in this manual include the following:

Placeholder	Meaning
source	Specifies the drive, directory, file, or device that will be transferred to a specified destination or used as input to a command.
destination	Specifies the drive, directory, file, or device that *source* will be transferred to.
string	Specifies a group of characters to be treated as a unit. A string can include letters, numbers, spaces, or any other characters and is usually enclosed in double quotation marks. Some commands, such as FIND, work with strings of text.

CHAPTER 1

COMMANDS OVERVIEW

INTRODUCTION

The following section lists MS OS/2 *MULTIUSER* command-line commands and utilities by category; the categories are also described. Included are MS OS/2 *MULTIUSER* commands, batch commands, utilities, multiuser utilities, configuration commands, and device drivers.

COMMANDS AND UTILITIES BY CATEGORY

MS OS/2 *MULTIUSER* Commands

A command-line command is any word or phrase that you can type at a prompt in order to carry out an action. MS OS/2 *MULTIUSER* command-line commands are built into the command interpreter, CMD.

CHCP
CHDIR
CLS
COPY
DATE
DEL
DETACH
DIR
DPATH
ERASE
EXIT
KEYS
MKDIR
MOVE
PATH
PROMPT
RENAME
RMDIR
SET
START
TIME
TYPE
VER
VERIFY
VOL

MS OS/2 *MULTIUSER* Batch Commands

Batch commands are read by the command interpreter and processed during the execution of a batch program.

CALL
ECHO
ENDLOCAL
EXTPROC
FOR
GOTO
IF
PAUSE
REM
SETLOCAL
SHIFT

MS OS/2 *MULTIUSER* Utilities

A utility is a program that is included as part of MS OS/2 *MULTIUSER* but is separate from the system command interpreter. You start a utility by typing its name at the command prompt.

MS OS/2 *MULTIUSER* Utilities Supported

ANSI
ATTRIB
BACKUP
CACHE
CHKDSK
CMD
COMP
CREATEDD
DDINSTAL
DISKCOMP
DISKCOPY
E
EAUTIL
FDISK
FIND
FORMAT
HELP
LABEL
MODE
MORE
PATCH
PRINT
PSTAT
RECOVER
REPLACE
RESTORE
SORT
SPOOL
SYSLOG
TRACE
TRACEFMT
TREE
UNPACK
XCOPY

Utilities Not Supported

- APPEND
- ASSIGN
- BOOT
- BREAK
- COMMAND
- GRAFTABL
- JOIN
- KEYB
- PICICHG
- PICPRINT
- PICSHOW
- SETCOM40
- SUBST

MS OS/2 *MULTIUSER* Utilities with Multiuser Functions

These multiuser utilities are invoked like commands. They enable the configuration and operation of the system's multiuser environment.

- CHANGE
  - AUDIT
  - EVENTS
  - LIMITS
  - PRNMODE
  - TERMINAL
- CONFIG
  - ACCESS
  - GROUP
  - PROFILES
  - SYSTEM
  - TERMINAL
  - USER
- CONNECT
- DISCONN
- EVENTS
- KILL
- LOGOUT
- MSG
- OWNER
- PASSWORD
- QUERY
  - AUDIT
  - EVENTS
  - HOST
  - LIMITS
  - LOGIN
  - MEMORY
  - PRINT
  - PROCESS
  - SESSION
  - TERMINAL
  - USER
- REGISTER
- RESERVE
- RESET
  - AUDIT
  - EVENTS
  - LIMITS
  - TERMINAL
- SHUTDOWN

MS OS/2 *MULTIUSER* Configuration *(CONFIG.SYS)* Commands

When you start MS OS/2 *MULTIUSER*, your system reads configuration commands from the *CONFIG.SYS* file on your start up disk.

MS OS/2 *MULTIUSER* Configuration Commands Supported

- AUTOFAIL
- BUFFERS
- CACHE
- CALL
- CODEPAGE
- COUNTRY
- DEVICE
- DEVINFO
  - KBD
  - PRN
  - SCR
- DISKCACHE
- HOSTNAME
- IFS
- IOPL
- LIBPATH
- LOG
- MAINTENANCE
- MAXWAIT
- MEMMAN
- PAUSEONERROR
- PRIORITY
- PROTECTONLY
- REBOOT
- REM
- RESOURCE
- RUN
- SECURITY
- SET
  - COMSPEC
  - DPATH
  - PATH
- SWAPPATH
- THREADS
- TIMESLICE
- TRACE
- TRACEBUF

Configuration Commands Not Supported

- BREAK
- FCBS
- PROTSHELL
- RMSIZE
- SHELL

MS OS/2 *MULTIUSER* Device Drivers

A device driver enables MS OS/2 *MULTIUSER* to handle a device that is installed on your system. This reference contains descriptions of the following supported drivers, which are included on your installation diskettes:

Device Drivers Supported by MS OS/2 *MULTIUSER*

COM0x.SYS
EXTDSKDD.SYS
VDISK.SYS

Device Drivers Not Supported by MS OS/2 *MULTIUSER*

ANSI.SYS
MOUSE.SYS
MSxxxx0x.SYS
PMDD.SYS
POINTDD.SYS

SECURITY CLASS RESTRICTIONS

As discussed in the *Citrix MULTIUSER User's Guide* and *Citrix MULTIUSER System Administrator's Guide*, there are four security classes:

- Administrator
- Operator
- User
- Guest

The Guest class has the lowest and Administrator the highest security clearance. For example, if the system allows a User class to have access to a certain function, an Operator or Administrator will also have access.

The DATE command is an example of a User function that has a security class restriction. A Guest or a User may query the date but may not change the date. An Operator or Administrator can query and change the date.

In this reference, each command and utility that has this consideration has a documentation heading, "Security Class Restrictions," where the appropriate details for that command are described. The following is a list of the commands and utilities that have security class restrictions:

CACHE
CHANGE AUDIT
CHANGE EVENTS
CHANGE LIMITS
CHANGE TERMINAL
CHKDSK
CONFIG ACCESS
CONFIG GROUP
CONFIG PROFILES
CONFIG SYSTEM
CONFIG TERMINAL
CONFIG USER
CONNECT
DATE
DDINSTAL
DISCONN
EVENTS
FDISK
FORMAT
KILL
LABEL
LOGOUT
MODE
OWNER
PASSWORD
PSTAT
QUERY AUDIT
QUERY EVENTS
QUERY LIMITS
QUERY LOGIN
QUERY MEMORY
QUERY PRINT
QUERY PROCESS
QUERY SESSION
QUERY TERMINAL
QUERY USER
RECOVER
REGISTER
RESERVE
RESET AUDIT
RESET EVENTS
RESET LIMITS
RESET TERMINAL
SHUTDOWN
SPOOL
SYSLOG
TIME
TRACE

Commands and utilities also access system resources like devices, directories, and files. If you attempt to execute a function that requires a resource you do not have access to, the function will fail.

For example, if you try to TYPE a file you do not have access to, the file will not be displayed on the screen.

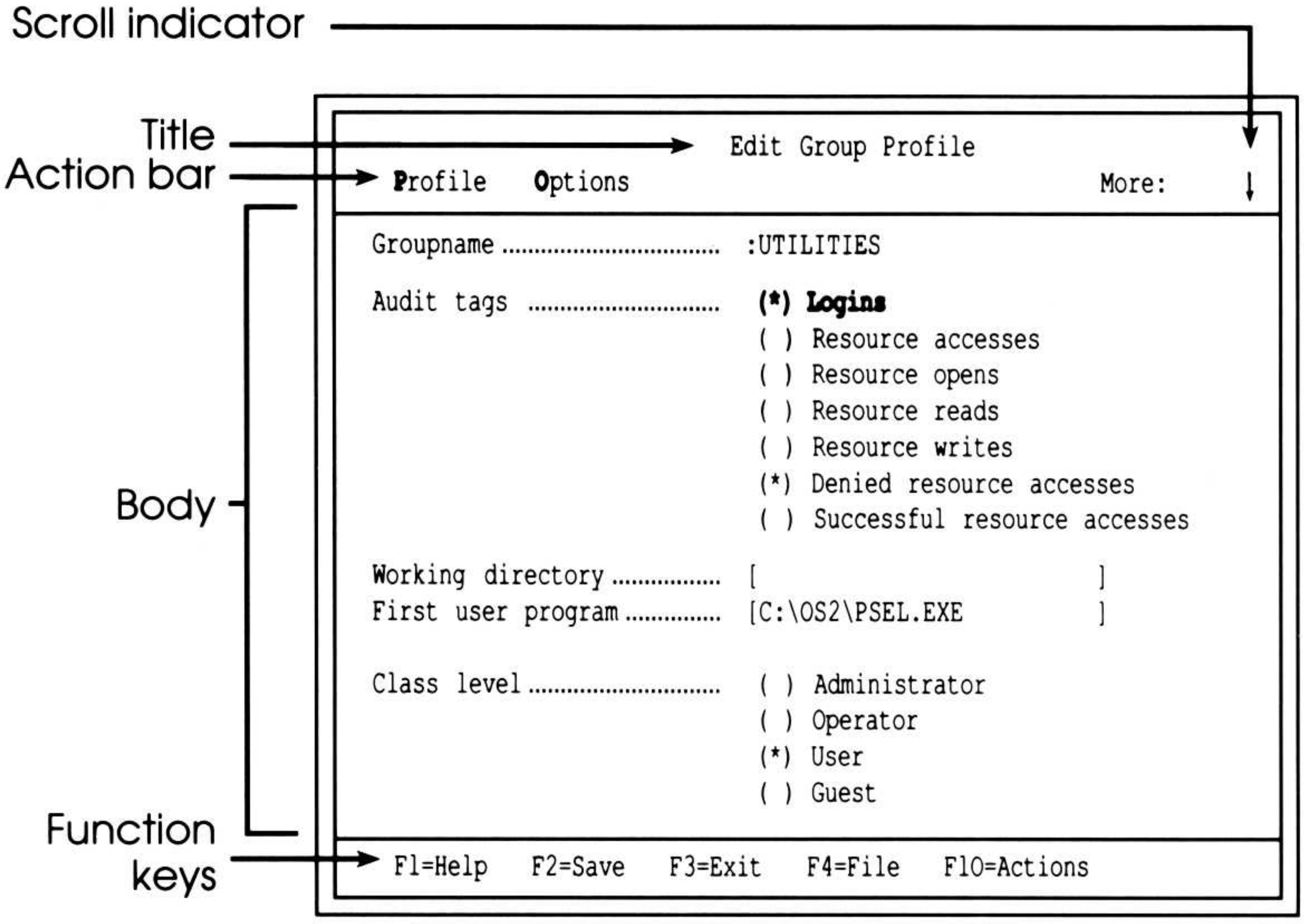
Many commands and utilities require access to resources that may be protected by Access Control Lists (ACLs). A separate list of the commands is not required since it is not the command that is secured, but the resource the command is seeking. Any further information about ACL protected resources may be found in the description of the command or utility.

FULL SCREEN INTERFACE

Many of the MS OS/2 *MULTIUSER* utilities can be run in a full screen mode with a user interface that presents a consistent easy-to-use method of interacting with the utility. This section is designed to familiarize you with the keystrokes used to interact with the full screen utilities. Navigating through the full screen interface is very intuitive and once you are familiar with a few of the basic keystrokes, you can easily use any of the full screen utilities.

Navigating Through the Full Screen Utilities

The full screen interface presents information on the screen in windows. A window is a rectangular region of the screen within which you can interact with a utility. A window can cover a portion of the screen or the entire screen. Windows can be broken down into different parts, although each window does not have to contain every part. A window can consist of a subset of the parts. See the diagram below for a description of the different window parts:



```
                              Edit Group Profile                    ↓
 Profile   Options                                          More:   ↓

 Groupname ........................ :UTILITIES

 Audit tags ....................... (*) Logins
                                    ( ) Resource accesses
                                    ( ) Resource opens
                                    ( ) Resource reads
                                    ( ) Resource writes
                                    (*) Denied resource accesses
                                    ( ) Successful resource accesses

 Working directory ................ [                         ]
 First user program ............... [C:\OS2\PSEL.EXE          ]

 Class level ...................... ( ) Administrator
                                    ( ) Operator
                                    (*) User
                                    ( ) Guest

  F1=Help   F2=Save   F3=Exit   F4=File   F10=Actions
```

The body of the window contains the information you are interacting with. If not all information can fit in the window, the information can be scrolled. Indicators at the top right corner (More ↑ ↓) of the window show whether there is more information not currently displayed in the window.

Some common keystrokes are used to navigate through the different areas of a window and to perform common functions. Utilities may have unique functions that they perform and may define function keys not used by other utilities to perform these functions. In any case, the function keys are listed at the bottom of the window with a brief (usually one word) description of what the key does. Also, you can always get Help in full screen utilities by pressing the **F1** function key.

The following is a list of common keys used in the full screen interface and what action or function is performed by each key:

DIRECTION keys — When you are in a list of choices the **DIRECTION** keys are used to move among the choices. In an input field, the **LEFT** and **RIGHT** arrow keys are used to move back and forth through the data in the field. The **TAB** and **BACKTAB** are the primary keys for moving between fields in a window.

ENTER — Selects or executes the highlighted choice in a menu or saves changes made in a popup.

ESC — Cancels the current operation or exits the popup without saving any changes.

SPACEBAR — Selects the highlighted item in a list of items. A tag is placed next to the selected item to indicate its state. In a list of items where more than one item can be selected, the **SPACEBAR** toggles an item between selected and not selected.

TAB

Moves the cursor to the field that is right of the current field or below the current field if the current field is the rightmost.

BACKTAB

Moves the cursor to the field that is left of the current field or above the current field if the current field is the leftmost.

ALT+F6

Toggles the cursor between a Help window and the field where the cursor was before the Help window was displayed. For this key to work, the Help window cannot be at the top of the field.

F1

Displays Help for the field or menu choice that is currently highlighted.

F2

Saves changes made when the main window of a utility is displayed.

F3

Exits the utility without saving changes.

F4

Saves changes made when the main window of a utility is displayed, then exits the utility.

F10

Toggles the cursor between the body of the window and the action bar. The **ALT** key by itself performs the same action.

CHAPTER 2

COMMANDS AND UTILITIES

ANSI (Extended Display and Keyboard Support)

Description

Changes or displays the support for ANSI escape sequences in an MS OS/2 *MULTIUSER* session.

Syntax

ANSI [ON | OFF]

Remarks

The default setting is ON. If you type ANSI by itself, the utility displays the current setting.

ATTRIB (File Attributes)

Description

Changes or displays the flags of the file you specify. The flags determine whether the file is read-only and whether it is affected when you use the BACKUP, RESTORE, and XCOPY utilities. For more information, see the individual utility descriptions.

Syntax

ATTRIB [+R | -R] [+A | -A] [*drive*:][*path*]*filename* [/S]

Parameters

+R
: Makes the file read-only.

-R
: Allows the file to be changed or deleted.

+A
: Sets the archive flag of the file.

-A
: Clears the archive flag of the file.

filename

Specifies the file for which you want to change the flags. You can use wildcard characters to affect more than one file.

Options

/S

Changes the flags of all files named *filename* in subdirectories as well as in directories.

Remarks

To see the flag settings of a particular file, type only ATTRIB and the *filename*. If the letter A appears in the listing, the archive flag is set for the file; if R appears in the listing, the read-only flag is set. You must have Read (R) permission to query the attribute and Attribute (A) permission to change the attribute flag.

Example

To make all the files read-only in the directory \SCHEDULE\MEETINGS on drive C and in all its subdirectories, type the following:

```
ATTRIB +R C:\SCHEDULE\MEETINGS\* /S
```

BACKUP (Save Files)

Description

Makes backup copies of files from one disk and stores them on another. You can use the RESTORE utility to copy these files back to their original location.

Syntax

BACKUP *drive1*:[*path*][*filename*] *drive2*: [/S] [/M] [/A] [/F[:*size*]] [/D:*date* [/T:*time*]] [/L:[*path*]*logfile*]

Parameters

drive1:

Specifies the drive that contains the files you want to backup.

filename

Specifies a single file you want to backup. You can use wildcard characters to make backup copies of a group of files with similar names.

drive2:

Specifies the drive that contains the disk on which you want to store the backup files. The BACKUP utility places the files in the root directory of this disk. If you choose to put the backup copies onto a fixed disk, BACKUP automatically creates a directory named BACKUP as a subdirectory of the root directory and places the files there.

Options

/S

Makes backup copies of the contents of all the subdirectories.

/M

Makes backup copies of only the files that have been changed since the last backup operation and turns off the archive flag of the original file. This option also makes backup copies of directories if their extended attributes have changed. Previous backup copies are deleted unless you also specify the /A option.

/A

Adds the backup files being created to the backup files on the destination disk without deleting the files that are already there. This option does not work if any of the files on the destination disk are backup copies made by using the BACKUP utility from MS-DOS version 3.21 or earlier.

/F[:*size*]

Formats the destination disk if it is unformatted. This option does not format a fixed disk nor does it format a disk that is already formatted. You can specify the memory capacity of the disk in kilobytes (or for a 1.2-megabyte or 1.44-megabyte disk, in megabytes) by specifying size. The easiest way to specify size is by the number of kilobytes of memory on the disk (360, 720, 1200, or 1440), but you can include the unit of measurement (for example, 360K, 360KB, 1.2M, 1.2MB) if you prefer. The BACKUP utility formats an unformatted destination disk even if you do not specify the /F option.

/D:*date*

Makes backup copies of only the files that were changed on or after the date specified.

/T:*time*

Makes backup copies of only the files that were changed at or after the time specified. Do not use this option without the /D:*date* option.

/L[:*logfile*]

Creates a log file in the root directory of the source disk or drive and puts a log of the backup operation in that file. You can specify a different drive or directory for the log file. If you do not specify *logfile*, the BACKUP utility names the file *BACKUP.LOG.*

Remarks

The backup files are stored in the *BACKUP.nnn* and *CONTROL.nnn* files; *nnn* is the number of the disk (starting from 001). As each disk is filled, the BACKUP utility prompts you for the next disk.

The source and destination disks do not have to be the same type; you can back up files from a fixed disk to a floppy disk or from one kind of floppy disk to another. If you do not specify the /A option, the BACKUP utility erases all the files that are already on the destination disk.

The BACKUP utility preserves the extended attributes and security attributes of the files and directories that it copies.

The BACKUP utility does not make backup copies of the following files: *OS2LDR, OS2KRNL, IBMBIO.COM, IBMDOS.COM, MSDOS.SYS, IO.SYS, COMMAND.COM,* and *CMD.EXE.* It also does not make backup copies of files that are in use on your start-up drive; to make a complete backup of your start-up drive, you must run BACKUP from the Installation disk.

You must have Read (R) permission on the source files and directories and Create (C) permission on the destination files.

See Chapter 6, "System Maintenance" in the *Citrix MULTIUSER System Administrator's Guide* for more information on backing up system data.

Examples

If you want to make backup copies of all the files in the \FILM\CRITIQUE directory on drive C and store them on an unformatted disk in drive A, type the following:

```
BACKUP C:\FILM\CRITIQUE A: /F
```

CACHE (HPFS Caching)

Description

Directs the system to load the cache driver for the High-Performance File System (HPFS). The CACHE utility is used to turn lazy writing on or off and to set or display the lazy writing parameters.

Syntax

CACHE [/MAXAGE:*time*] [/DISKIDLE:*time*] [/BUFFERIDLE:*time*]

or

CACHE [/LAZY:ON | /LAZY:OFF]

Parameters

/MAXAGE:*time*
Sets the maximum amount of *time* in milliseconds that a cache block can store information before the system writes that information to the disk. The default time is 5000 milliseconds.

/DISKIDLE:*time*

Sets the maximum amount of *time* in milliseconds that the disk can be idle before the system writes unsaved information in the cache blocks to the disk. The default time is 1000 milliseconds. This option affects only cache blocks that have unsaved information in them for the amount of time specified by /BUFFERIDLE.

/BUFFERIDLE:*time*

Sets the maximum amount of *time* in milliseconds that a cache block can store information before it becomes subject to the effect of the /DISKIDLE option. The default time is 500 milliseconds.

/LAZY:ON | /LAZY:OFF

Turns lazy writing ON or OFF for all disks or partitions that are formatted for HPFS. The default setting is /LAZY:ON.

Remarks

The CACHE utility can be run from the command line or it can be started with *CONFIG.SYS* using the RUN command. MS OS/2 *MULTIUSER* inserts this in *CONFIG.SYS* as "RUN=C:\OS2\CACHE.EXE."

On disks or partitions formatted for HPFS, the system can store information in temporary 2 KB cache blocks — where the system can locate it faster — rather than writing the information directly to the disk. The CACHE options determine when the information in the cache block is written to the disk. If lazy writing is not turned on, the system writes information directly to the disk rather than storing it in cache blocks.

If you use the CACHE utility to turn lazy writing on, the prompt disappears in the session in which you are running the utility. To avoid this, use the START command to create a new session to be used only for running the CACHE utility.

If you type CACHE by itself, the utility displays its current setting.

Security Class Restrictions

This utility can be executed only by an Administrator or Operator class.

Examples

To start a new session and run the CACHE utility, turning lazy writing on, type the following:

```
START CACHE /LAZY:ON
```

To start lazy writing for the system option in *CONFIG.SYS*, type the following:

RUN=C:\OS2\CACHE.EXE [*parameters*]

CALL (Nest Batch Files)

Description

Calls one batch file from another. You use CALL in a batch file in order to run another batch file and then return to the first one. Although CALL can be used from the command prompt, it is designed to be placed in a batch file.

Syntax

CALL [*drive*:][*path*]*batchfile* [*arg*]

Parameters

batchfile
Specifies the name of the batch file (without extension) that you want to call from within another batch file.

arg
Specifies the arguments for the batch file being called.

Remarks

When you use the CALL command, MS OS/2 *MULTIUSER* treats everything in the batch file you call as though it were a single command; once that command has been carried out (that is, once the second batch file has been run), the system resumes running the first batch file where it was interrupted.

Do not use pipes or redirection symbols with the CALL command.

Example

To run the file *CHECKNEW.CMD* from another MS OS/2 *MULTIUSER* batch file and pass it the /T option, include the following line in the first batch file:

```
CALL CHECKNEW /T
```

CHANGE (Configure Current State)

Description

The CHANGE utilities are used to modify current settings. Any changes made are temporary. If the changes are for a user login, the changes last until the user logs out. Other changes last until the next system restart.

Syntax

The command "CHANGE" can invoke any one of the change utilities. The command line format is:

```
CHANGE [AUDIT | EVENTS | LIMITS | PRNMODE | TERMINAL]
[/?]
```

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

Modifications done by the CHANGE utilities do not have any effect on any of the profiles. These changes are only in effect until the object that was changed is reset or restarts. For example, in the case of a user, the changes are in effect until the user logs off.

This utility changes the runtime configuration parameters without affecting the permanent configuration parameters. To change the permanent configuration parameters, use the CONFIG utility.

CHANGE AUDIT

Description

CHANGE AUDIT will temporarily change the system-wide security audit events that are being logged.

Syntax

CHANGE AUDIT [/G+:*accessletters*] [/G-:*accessletters*] [/D+:*accessletters*] [/D-:*accessletters*] [/?]

Options

/G+:*accessletters*
Specifies to add this list of access types (*accessletters*) to those being audited when access is granted.

/G-:*accessletters*
Specifies to remove this list of access types (*accessletters*) from those being audited when access is granted.

/D+:*accessletters*
Specifies to add this list of access types (*accessletters*) to those being audited when access is denied.

/D-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is denied.

accessletters

These are equivalent to the access permissions established using CONFIG ACCESS. See the *Citrix MULTIUSER System Administrator's Guide* for a complete description of these permissions.

*	All
R	Read
W	Write
C	Create
D	Delete
X	Execute
A	Change attributes
U	Use
V	Reserve
L	Login
S	Query self
O	Query other
M	Modify self
T	Modify other

/? (help)

Displays the syntax for the utility and information about the utility's options.

Remarks

The audited events are established using the CONFIG SYSTEM utility and are placed in the system profile. When the system is started, the current audit logging state is taken from the system profile. At any time during system operation, the selection of audit events being logged can be changed using CHANGE AUDIT. This change will only be in effect until the system is restarted.

This only changes the system-wide auditing flags. Audit flags can also be placed on users and resources. The commands, CONFIG USER and CONFIG ACCESS, must be used to change the user and resource audit flags.

Auditing can be turned on and off using CHANGE EVENTS. If auditing is off, the audit flags established here will not be used until audit event logging is turned on.

Use this command with caution. Auditing system-wide granted access of many access types can generate a great amount of logging activity and can severely impact system performance. It is far more common to audit only denied accesses.

Security Class Restrictions

This utility is available only to an Operator or Administrator class.

Example

```
CHANGE  AUDIT  /D+:R  /G-:*
```

This removes all (*) auditing of granted accesses and adds auditing of denied Read (R) accesses.

CHANGE EVENTS

Description

CHANGE EVENTS will temporarily change which events are being logged. This is effective only until system shutdown.

Syntax

CHANGE EVENTS [/SET:*eventletters*] [/CLR:*eventletters*] [/?]

Options

/SET:*eventletters*

Turns on logging of these events. The event categories already being logged remain on.

/CLR:*eventletters*

Turns off logging of these events.

eventletters

A list of letters, each of which denotes a specific system event category. These letters are:

A	Security access audit
E	Errors
M	Multiuser events
P	Profile update events
R	Resource use events
S	Security secondary audit events
T	Terminal subsystem events

/? (help)

Displays the syntax for the utility and information about the utility's options.

Remarks

This sets or clears events that are being logged but only until the next reboot. To permanently change the event log selection, see CONFIG SYSTEM.

The events specified are added (/SET) or removed (/CLR) from the already existing list.

At least one option, /SET or /CLR must be specified.

Security Class Restrictions

This utility is available only to an Operator or Administrator class.

Example

CHANGE EVENTS /SET:ET

Adds the two events (E and T) to the categories currently being logged.

CHANGE LIMITS

Description

CHANGE LIMITS temporarily changes the resource limits for one or more users in the system. If no *loginname* is specified on the command line, the current *loginname* is used.

Syntax

CHANGE LIMITS [*username*[*.groupname*]] [/MIN*resource*:*n*] [/MAX*resource*:*n*] [/?]

Parameter

username[*.groupname*]
: Identifies the *loginname* of the user whose resource limits are to be changed. If the *groupname* is not specified, all groupnames for that *username* are used (equivalent to *username.**). The default for the CHANGE LIMITS command is to change the user resource limits for the current *loginname.*

Options

/MIN*resource*:*n* and /MAX*resource*:*n*
: Specifies what resource minimum or maximum limit is to be changed. More than one of these can be specified, but at least one must be specified.

n
A new minimum or maximum limit.

The following specific resource limit options are available for /MIN*resource*:*n* and /MAX*resource*:*n*.

/MINMEMORY:*n*
/MAXMEMORY:*n*
/MINTHREADS:*n*
/MAXTHREADS:*n*
/MINFILES:*n*
/MAXFILES:*n*
/MINSEMAPHORES:*n*
/MAXSEMAPHORES:*n*
/MAXLOGINS:*n*
/MAXSESSIONS:*n*

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

If RESOURCE=OFF in *CONFIG.SYS*, the message "Resource management is turned off" is displayed.

Security Class Restrictions

This utility is available only to an Operator or Administrator.

Examples

The *loginname* can contain wildcard characters as in the following examples, provided the user has the proper security classification:

```
CHANGE LIMITS  /MAXTHREADS:25  *.*
CHANGE LIMITS  /MAXTHREADS:25  *.WRKGRP
CHANGE LIMITS  /MAXTHREADS:25  MIKEH.WRKGRP
```

CHANGE PRNMODE

Description

CHANGE PRNMODE changes the line and character spacing for a parallel printer.

Syntax

CHANGE PRNMODE *portname* [/LPI:[6 | 8]] [/CPL:[80 | 132]] [/?]

Parameter

portname
Specifies the parallel port the printer is connected to.

Options

/LPI:[6 | 8]
Specifies vertical spacing (the number of lines per inch). This number can be either 6 or 8.

/CPL:[80 | 132]
Specifies the number of characters per line. This number can be either 80 or 132.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

This change is for the current loginname. Therefore, it is effective only for the loginname of the user who issues the command. CHANGE PRNMODE will only be effective if the device is being spooled by the spooler.

To change the printer mode for the entire system, refer to the MODE command.

Example

To set LPT1 to 132 characters per line:

```
CHANGE PRNMODE LPT1 /CPL:132
```

CHANGE TERMINAL

Description

CHANGE TERMINAL temporarily changes a specified terminal in the system.

Syntax

CHANGE TERMINAL *terminalname*
[/TERMTYPE:*terminaltype*] [/CONNECT+:*conn1,conn2,...*]
[/CONNECT-:*conn1,conn2,...*] [/FLOW+:*flow1,flow2,...*]
[/FLOW-:*flow1,flow2,...*] [/PARITY:*value*] [/BAUD:*value*]
[/STOP:*value*] [/DATA:*value*] [/?]

Parameter

terminalname
The *terminalname* is the name given to the terminal by the System Administrator when the terminal was configured. QUERY TERMINAL can be used to determine the *terminalname.*

Options

/TERMTYPE:*terminaltype*
Redefines the terminal type that is connected to *terminalname.*

/CONNECT+:*conn1,conn2,...*
Specifies to add the list of connect settings to terminal profile *terminalname.* Valid connect settings are:

CTS	Connect on CTS
DSR	Connect on DSR
RING	Connect on ring indicator
DCD	Connect on data carrier detect
CHAR	Connect on first character
BRK	Disconnect on break
*	All

/CONNECT-:*conn1,conn2,...*
Specifies to remove the list of connect settings from terminal profile *terminalname.* Valid connect settings are defined above.

/FLOW+:*flow1,flow2,...*
Specifies to add the list of flow settings to terminal profile *terminalname.* Valid flow settings are:

XON	XON/XOFF enable
DUP	XON/XOFF full duplex
RTS	RTS/CTS enable
RTSH	RTS input handshaking
DTR	DTR/DSR enable
DTRH	DTR input handshaking
CTSH	CTS output handshaking
DSRH	DSR output handshaking
DCDH	DCD output handshaking
DSRS	DSR input sensitivity
XPC	XPC protocol (used with XON)
*	All

/FLOW-:*flow1,flow2,...*

Specifies to remove the list of flow settings from terminal profile *terminalname*. Valid flow settings are defined above.

/PARITY:*value*

Redefines the parity setting of the terminal to *value*. Valid parity settings are 0 (no parity), 1 (odd), and 2 (even).

/BAUD:*value*

Redefines the baud rate to *value*. Valid baud rates are 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, 19200, and 38400.

/STOP:*value*

Redefines the number of stop bits to *value*. Valid stop bit values are 1 and 2.

/DATA:*value*

Redefines the number of data bits to *value*. Valid data bit values are 7 and 8.

/? (help)

Displays the syntax for the utility and information about the utility's options.

Remarks

CHANGE TERMINAL is used to change the terminal type of the terminal for the duration of the login. It does not change the permanent configuration that is stored on fixed disk. To change the permanent configuration, use the CONFIG TERMINAL utility.

Changing the terminal type is provided primarily to allow for dial-in terminals. A terminal may be configured as a generic dial-in terminal which means that the terminal driver uses a subset of the available functions that are provided by most terminals. After login, a user could use the CHANGE TERMINAL utility to install the driver that matches the specific terminal being used. This allows the user to take advantage of the specific capabilities of that terminal.

Security Class Restriction

An Administrator or Operator can use the CHANGE TERMINAL command to change the terminal characteristics for any terminal in the system.

A User or Guest may only use the CHANGE TERMINAL command to change the terminal characteristics for the terminal on which he or she is logged into.

Example

The following is an example of how to change the terminal type:

```
CHANGE TERMINAL TERM01 /TERMTYPE:WYSE150
```

CHCP (Change Code Page)

Description

Switches to the specified system code page, which must have been defined previously in your *CONFIG.SYS* file.

Syntax

CHCP [*nnn*]

Parameter

nnn
Specifies the code page to be used. This argument must be a three-digit number from the list under the codepage command.

Remarks

If you type CHCP by itself, MS OS/2 *MULTIUSER* displays the number of the active code page.

You can use this command only if code pages have been previously prepared by using the CODEPAGE configuration command.

Any program that you run after starting a new code page uses the new code page. Programs that started running before you started the new code page still use the original code page.

Code pages are supported at the console for both keyboard code pages and video code pages fully based upon the parameters that are defined on the DEVINFO, the COUNTRY and the CODEPAGE *CONFIG.SYS* commands.

Code page support for terminals is limited to keyboard code pages. Unlike video code page support, the keyboard code page support requires no underlying hardware to aid in the keystroke translation process; therefore, hardware-independent keyboard code page support is provided at terminals. When a request is made to change the keyboard code page, the system will carry out the request and begin translating keystrokes using the designated code page for the country that is defined on the "COUNTRY =" line within the *CONFIG.SYS* file.

Video code pages are not supported at serial connected terminals; however, if a request is made to change the codepage, the system will honor the request. Further video output that is dependent on the code page may not appear on the screen correctly.

Example

To switch the code page to 850 (multilingual), type the following:

```
CHCP 850
```

CHDIR (Change Directory)

Description

Changes the current directory. You can abbreviate CHDIR as CD.

Syntax

CHDIR [*drive*:][*path*]

Remarks

If you type CHDIR by itself, MS OS/2 *MULTIUSER* displays the name of the current directory of the current drive.

You cannot use CHDIR to change drives but you can use it to change the current directory of another drive. If you then switch to that drive, you will be in the directory you specified. Typing CHDIR plus the letter of another drive displays the name of the current directory on that drive.

You cannot change to a directory unless you have Read (R) access permission.

Examples

To change from your current directory to the ENTREE subdirectory, type the following:

```
CD ENTREE
```

To change from your current directory to its parent directory, type the following:

```
CD ..
```

To return to the root directory, type the following:

```
CD \
```

If your current directory is POETRY and its parent directory is BOOKS, and you want to change to the FICTION directory, which is also under BOOKS, type the following:

```
CD ..\FICTION
```

CHKDSK (Analyze Disk)

Description

Checks a disk or partition for errors and displays a summary of how space is used on that disk or partition. The CHKDSK utility also displays the volume label, the volume serial number, and the type of file system being used by the disk or partition.

Syntax

CHKDSK [*drive*:][*path*][*filename*] [/C] [/F[:*n*]] [/V]

Parameters

drive:
Specifies the disk drive (partition) to check. If you do not specify *drive*, the CHKDSK utility checks the current drive (partition).

filename
(FAT file system only) Specifies the file to check. You can use wildcard characters to specify a group of files. The CHKDSK utility reports how many of the specified files are stored in noncontiguous sectors.

Options

/C

(HPFS only) Specifies that the CHKDSK utility is to correct errors only if the file system is in an inconsistent state when you restart your computer; if the file system is in a consistent state, CHKDSK simply terminates, regardless of what other options you have specified. (The file system may be left in an inconsistent state if the system crashes or is restarted while there are open files or while the directory structure is being updated.) You can specify the level of data recovery by using the /F:*n* option.

/F[:*n*]

Specifies that the CHKDSK utility is to correct errors it finds on the disk. If you do not specify this option, CHKDSK does not correct any errors. The *n* argument is available only for HPFS. It specifies the level of data recovery and must be a number in the range 0 through 3; the default level is 2. Level 0 is the same as not specifying /F at all. At level 1, CHKDSK corrects inconsistencies, moves data in reserved sectors to the file system data sectors, and updates all references to the sectors in file-system structures.

These actions may create orphaned disk space — that is, space once occupied by corrupt files removed by CHKDSK but still marked "in use" in the disk-usage map. At level 2, CHKDSK does everything included at level 1 and also recovers any recognizable structures from orphaned disk space. At level 3, CHKDSK does everything included at level 2 and also recovers any recognizable structures from anywhere in the disk partition. The higher the level of recovery you specify, the more thorough the recovery will be, but the longer it will take.

/V

Displays the name of each file in each directory as it is checked.

Remarks

If you type CHKDSK by itself, the utility displays the status of the disk in the current drive. If you specify a *filename* but no *drive* (FAT file system only), CHKDSK displays the status of the disk in the current drive and of the individual file.

If you specify the /F option, the CHKDSK utility reports an error if there are any open files on the disk. If you do not specify /F and there are open files, CHKDSK may report that there are lost clusters on the disk. This happens when the disk's file-system table has not been updated regarding open files.

When the CHKDSK utility finds lost clusters on a disk, it displays a message asking you whether you want the lost clusters recovered. If you did not specify /F, answering **Y** (for yes) only shows what CHKDSK would have done if you had specified /F, and answering **N** (for no) does nothing. If you did specify /F, however, answering **Y** causes CHKDSK to recover the lost clusters and store them in files named *FILEnnnn.CHK*, where *nnnn* is a consecutive number beginning from 0001; answering **N** causes CHKDSK to erase the lost clusters. If there are extended attributes in the lost clusters, CHKDSK does not erase the lost clusters when you specify /F and answer **N**.

If CHKDSK finds errors in extended attributes, it places the recovered attributes in files named *EAnnnn.CHK*, where *nnnn* is a sequential number assigned by MS OS/2 *MULTIUSER*. You can look at these files by using the TYPE command. If you can identify which of your files the extended attributes belong to, you can join the extended attributes to the file by using the EAUTIL utility.

For fixed disks, the CHKDSK /F option should only be used in maintenance mode.

WARNING: Do not run an older version of the CHKDSK utility (earlier than MS OS/2 version 1.2) on a disk or partition that contains extended attributes. You will lose the extended attributes and you may corrupt the files on the partition.

Although you can use the CHKDSK utility to get information about any drive, you cannot use it to fix errors on the drive from which you started your system or the drive from which you are running CHKDSK. To check your start-up drive, you must run CHKDSK from the MS OS/2 *MULTIUSER* Install disk. You can do the same thing to check the drive from which you would normally run CHKDSK or you can copy the CHKDSK utility to another drive and run it from there.

For more information about interpreting CHKDSK messages and about fixing errors on your start-up disk, see Chapter 13, "System Maintenance" in the *Citrix MULTIUSER System Administrator's Guide.*

Security Class Restrictions

Administrator and Operator classes have permission to use the CHKDSK utility on fixed disk partitions. An error message is displayed if a user who is not authorized to do so attempts to use the CHKDSK utility on fixed disk partitions.

All classes can run CHKDSK on a diskette drive.

Example

To check and fix errors on drive C (your start-up drive), first SHUTDOWN the system. Insert the MS OS/2 *MULTIUSER* Install disk in drive A, press **CTRL+ALT+DEL**, insert disk 1 when prompted and press **ENTER**. When the first panel appears, press the **ESC** key and type the following at the command prompt:

```
CHKDSK C: /F
```

CLS (Clear Screen)

Description

Clears the screen, leaving only the prompt and the cursor.

Syntax

```
CLS
```

CMD (MS OS/2 *MULTIUSER* Command Interpreter)

Description

Starts a new MS OS/2 *MULTIUSER* command interpreter.

Syntax

CMD [*drive*:][*path*] [/S] [/C *command(s)* | /K *command(s)*]

Parameter

drive:path

Specifies the location of the new *CMD.EXE.* If you do not specify a *drive* and/or *path*, the system uses the command interpreter specified by the COMSPEC environment variable.

Options

/S

Instructs the new command interpreter not to set up the signal handler. For example, the command interpreter will not take any action on the **CTRL+C** and **CTRL+BREAK** hotkeys.

/C *command(s)*

Instructs the new command interpreter to perform the command or commands specified and then return control to the command interpreter that called it. If you specify multiple commands, separate them with an ampersand (**&**).

/K *command(s)*
Instructs the new command interpreter to perform the command or commands specified and then continue to run. If you specify multiple commands, separate them with an ampersand (&).

Remarks

If you type CMD by itself, the system starts the new command interpreter in the current session.

When you start a command interpreter, you also create a command environment. This environment is a copy of the environment from which you started the command interpreter. You can change this new environment without affecting the old environment.

For an alternate way to run CMD and have it start other commands, see the START command.

Examples

To start CMD and have it check the disk in drive B and then end, type the following:

```
CMD /C CHKDSK B:
```

To start CMD and have it automatically run a batch file, type the following:

```
CMD /K OS2INIT.CMD
```

where OS2*INIT.CMD* is a batch file.

COMP (Compare Files)

Description

Compares two files or sets of files to see whether they are the same.

Syntax

COMP [*drive*:][*path*][*filename1*] [*drive*:][*path*][*filename2*]

Parameters

filename1
Specifies the name of the first file.

filename2
Specifies the name of the second file. If *filename2* is the same as *filename1* except that the file is on a different drive, you can type only the drive letter of the second file.

Remarks

If you specify only a path or a drive without any *filename*, the COMP utility assumes that you want to compare all the files in that directory or on that drive.

If you type COMP by itself, the utility prompts you for the *filename(s)*. You can use wildcard characters to specify a group of files.

If the files found in one directory differ in size from the files found in the other directory, COMP displays a message asking if you want to continue. If the files are the same size but their contents differ, COMP automatically displays the location and contents of each nonmatching byte.

You must have Read (R) access permission on both files being compared.

Examples

To compare each file with the extension *.DOC* in the current directory on drive C with each file of the same name with the extension *.BAK* on drive B, type the following:

```
COMP C:*.DOC B:*.BAK
```

To compare the files in the root directory on drive A with the contents of the TEMP directory on drive C, type the following:

```
COMP A: C:\TEMP
```

To compare the contents of the SALES directory with the current directory, type the following:

```
COMP \SALES
```

CONFIG (Configure Profile Permanently)

Description

The CONFIG utilities are used to manage profiles for the system, groups, users, terminals, and Access Control Lists (ACLs).

Syntax

```
CONFIG  [ACCESS | GROUP | PROFILES | SYSTEM |
TERMINAL | USER] [/?]
```

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

If no parameters are specified on the command line for the CONFIG command, a full screen menu is displayed allowing profiles, terminals, or access to system resources to be configured. See Chapter 4, "Introduction to the CONFIG Utility" in the *Citrix MULTIUSER System Administrator's Guide* for an illustration of the CONFIG menu.

Profile changes that are saved by the CONFIG utilities will automatically update user logins or terminals that are currently active.

CONFIG ACCESS

Description

This utility configures system resources with security attributes, consisting of Access Control Lists (ACLs) and Audit Masks.

Syntax

```
CONFIG ACCESS
```

Remarks

This provides a full screen interface for query and modification of the security attributes.

The functions that can be performed with the full screen version of CONFIG ACCESS can also be performed from the command line. See Appendix A, "Command Line Invocation of the CONFIG Utility."

See Chapter 7 of the *Citrix MULTIUSER System Administrator's Guide* for an illustration of the CONFIG ACCESS full screen presentation.

Security Class Restrictions

Administrators can view and change all security attributes. Operators can view all security attributes but can only change permissions for the files they own. Users and Guests can view security attributes only for those resources to which they have access, and can change only those that they own.

CONFIG GROUP

Description

Configures a group profile.

Syntax

CONFIG GROUP [*groupname*]

Parameter

groupname
A *groupname* in the MS OS/2 *MULTIUSER* system.

Remarks

The CONFIG GROUP utility displays a full screen window of the group profile specified by *groupname*. If no *groupname* is specified, a list of the *groupnames* is displayed so one can be selected. Use the CONFIG PROFILES command to create a new *groupname*.

The functions that can be performed with the full screen version of CONFIG GROUP can also be performed from the command line. See Appendix A, "Command Line Invocation of the CONFIG Utility."

See Chapter 5 of the *Citrix MULTIUSER System Administrator's Guide* for an illustration of the CONFIG GROUP full screen presentation.

Security Class Restrictions

Administrator class has full access to all the CONFIG functions. Operator class can view all configurations but cannot modify any. User and Guest classes can view only their user profile and related group defaults; they can modify nothing.

CONFIG PROFILES

Description

Configures the system, group, and user profiles.

Syntax

CONFIG PROFILES

Remarks

The CONFIG PROFILES utility displays a window of group profiles and the users who are members of the groups. From the window any of the system, group, or user profiles can be created, modified, or deleted.

See Chapter 5 of the *Citrix MULTIUSER System Administrator's Guide* for an illustration of the CONFIG PROFILES full screen presentation.

Security Class Restrictions

Administrator class has full access to all the CONFIG functions. Operator class can view all configurations but cannot modify any. User and Guest classes can view only their user profile and related group defaults; they can modify nothing.

CONFIG SYSTEM

Description

Configures the system profile.

Syntax

```
CONFIG SYSTEM
```

Remarks

The CONFIG SYSTEM utility displays a full screen window of the system profile. From this window any of the system profile settings can be modified.

The default hotkey definitions and autologin user are used when a new terminal profile is created. Changing these settings will not have any effect on existing terminal profiles.

The default first user program and resource limits are used when a new group profile is created. Changing these settings will not have any effect on existing group profiles. All other settings are system-wide and take effect when changes are saved.

The functions that can be performed with the full screen version of CONFIG SYSTEM can also be performed from the command line. See Appendix A, "Command Line Invocation of the CONFIG Utility."

See Chapter 5 of the *Citrix MULTIUSER System Administrator's Guide* for an illustration of the CONFIG SYSTEM full screen presentation.

Security Class Restrictions

Administrator class has full access to all the CONFIG functions. Operator class can view all configurations but cannot modify any. User and Guest classes do not have access to the system profile.

CONFIG TERMINAL

Description

Configures the terminal profiles.

Syntax

CONFIG TERMINAL *terminalname*

Parameter

terminalname

The *terminalname* is the name given to the terminal by the System Administrator when the terminal is configured. QUERY TERMINAL can be used to determine the *terminalname.*

Remarks

The CONFIG TERMINAL utility displays a window of terminals that are configured and the subsystem(s) to which terminals are attached. From this window, terminals and terminal subsystems can be created, modified, or deleted.

The functions that can be performed with the full screen version of CONFIG TERMINAL can also be performed from the command line. See Appendix A, "Command Line Invocation of the CONFIG Utility."

See Chapter 6 of the *Citrix MULTIUSER System Administrator's Guide* for an illustration of the CONFIG TERMINAL full screen presentation.

Security Class Restrictions

Administrator class has full access to all the CONFIG functions. Operator class can view all configurations but cannot modify any. User and Guest classes do not have access to terminal profiles.

CONFIG USER

Description

Configures the user profiles.

Syntax

CONFIG USER *username*[*.groupname*]

Parameter

username[*.groupname*]

Specified to complete user *loginname*. If the *groupname* is not specified, the default *groupname* for the *username* is used.

Remarks

The CONFIG USER utility displays a full screen window of the user profile specified by the *username* and the *groupname*. If no *username* is specified, a list of MS OS/2 *MULTIUSER* usernames and the groupnames they are members of is displayed. From this window any of the user profile parameters can be modified.

Group override settings for the user will default to the settings defined in the group profile when the user logs in, if they are left blank in the user profile. Group override settings include the working directory, first user program, class level, and resource limits.

The functions that can be performed with the full screen version of CONFIG USER can also be performed from the command line. See Appendix A, "Command Line Invocation of the CONFIG Utility."

See Chapter 5 of the *Citrix MULTIUSER System Administrator's Guide* for an illustration of the CONFIG USER full screen presentation.

Security Class Restrictions

Administrator class has full access to all the CONFIG functions. Operator class can view all configurations but cannot modify any. User and Guest classes can view only their user profile and related group defaults; they can modify nothing.

CONNECT (Get Disconnected Login)

Description

Attach a disconnected login to a terminal.

Syntax

CONNECT [*loginID*] [/T:*terminalname*] [/PASSWORD:*password*] [/F] [/V] [/?]

CONNECT [\ *username.groupname*\ *loginID*] [/T:*terminalname*] [/PASSWORD:*password*] [/F] [/V] [/?]

CONNECT [\ *terminalname*] [/T:*terminalname*] [/PASSWORD:*password*] [/F] [/V] [/?]

Parameters

loginID
Identifies a login.

\ *username.groupname*\ *loginID*
Identifies a specific login belonging to the specific *loginname.*

\ *terminalname*
Identifies all logins connected to *terminalname.*

Options

/T:*terminalname* (destination *terminalname*)
Connects the specified disconnected login to the terminal identified by *terminalname*, which must be an active terminal.

/PASSWORD:*password*
Specifies the *password* associated with the *loginname* that owns the specified *loginID.* Only needed in cases where the specified *loginID* belongs to a *loginname* that is different from the current *loginname.*

/F (force a login to move)
Connects the specified login to the specified terminal, even if the specified login is connected to a terminal.

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

If no *loginID* or *terminalname* is specified, CONNECT uses all disconnected logins of the current *loginname.* If *terminalname* is specified, it must identify an active terminal. Wildcards may be used in the input parameter specifying the *loginID.* If a single wildcard is present (*), it is interpreted as a wildcard for the *loginID* and specifies all *loginIDs* in the system.

CONNECT operates only on disconnected logins, unless the /F option is used.

Disconnected logins are attached to the disconnected terminal "DISC". The *terminalname* "DISC" may be used to specify all disconnected logins in the system. A *terminalname* other than "DISC" may be used with the /F option to force all logins at the specified terminal to move to the current terminal.

If no destination terminal is specified with the /T:*terminalname* option, CONNECT attaches the specified disconnected logins to the current terminal.

If the /PASSWORD option is not specified and the target *loginID* belongs to a *loginname* that is different from the current *loginname,* CONNECT will prompt for the *password.*

Security Class Restrictions

CONNECT may only be used to connect disconnected logins belonging to the current *loginname* to the current terminal, unless the current *loginname* has sufficient security classification to connect logins belonging to other *loginnames* as well as affect terminals other than the current terminal. Administrator class and Operator class have full access to all CONNECT functions. User class and Guest class may only use CONNECT functions on logins that belong to the current *loginname.*

Examples

example 1: no options

CONNECT attach all disconnected logins belonging to the current *loginname* to the current terminal.

```
CONNECT
```

example 2: *username*

CONNECT all disconnected logins belonging to *loginnames* with the *username* BRAD to the current terminal.

```
CONNECT \BRAD
```

example 3: *loginname*

CONNECT all disconnected logins with the *loginname* BRAD.WRKGRP to the current terminal.

```
CONNECT \BRAD.WRKGRP
```

example 4: loginID

CONNECT connects the disconnected login with *loginID* of 12 to the current terminal.

```
CONNECT 12
```

example 5: fully specified loginID

CONNECT connects the disconnected login with *loginID* 10 belonging to the *loginname* of BRAD.WRKGRP to the current terminal.

```
CONNECT \BRAD.WRKGRP\10
```

example 6: terminal option

CONNECT connects the disconnected login with *loginID* 10 to the terminal identified as SERIAL0.

```
CONNECT 10 /T:SERIAL0
```

example 7: force option

CONNECT connects the login with *loginID* 23 to the terminal identified as SERIAL0, regardless of where the login was attached.

```
CONNECT 23 /T:SERIAL0 /F
```

COPY (Copy Files)

Description

Copies information from a *source* to a *destination*. The *source* is usually a file or files, but it can also be a drive or directory or the output of a device (such as the keyboard); the *destination* can be one or more files or a device.

Syntax

COPY *source* [/A | /B] [+*source* [/A | /B] [...]] *destination* [/A | /B] [/V] [/F]

Parameters

source

Specifies where the information will be copied from. If the *source* is a file, MS OS/2 *MULTIUSER* assumes that the file is in the current directory on the current drive, unless you tell it otherwise by specifying a drive and path. The *source* can also be a drive or a directory. You can copy multiple files by using wildcard characters. You can copy multiple-source files to one destination file by specifying multiple-source filenames separated by plus signs (+). If the source is a *device*, the system takes the input from that device and copies it to the *destination.*

destination

Specifies where the information will be copied to. The *destination* can be a single file or you can use wildcard characters to specify a group of files. The *destination* can also be a directory or a device. If you specify only a drive as the *destination*, the system copies to the current directory on that drive; if you do not specify a drive, the system uses the current drive. If the *destination* is a file that does not already exist, the system creates a new file; if the file already exists, the system writes over the old file.

Options

/A

Treats the *source* or *destination* as ASCII text. This option applies to the *filename* preceding it and to all remaining *filenames* in the command until MS OS/2 *MULTIUSER* encounters a /B option, in which case the /B option applies to the *filename* that precedes it. When /A comes after the *source*, the system copies everything up to the first **CTRL+Z** character, which it interprets as an end-of-file mark, and does not copy anything after that. When /A comes after the *destination*, the system adds an end-of-file character as the last character of the *destination* file. When you are copying multiple files to one *destination*, the default option is always /A.

/B

Treats the *source* or *destination* as a binary file. This option applies to the *filename* preceding it and to all remaining *filenames* in the command until MS OS/2 *MULTIUSER* encounters an /A option, in which case the /A option applies to the *filename* that precedes it. When /B comes after the source, the system copies everything and does not interpret any **CTRL+Z** characters as end-of-file marks. When /B comes after the *destination filename*, the system does not add an end-of-file character to the new file.

/V

Checks whether the sectors the information was copied to can be read.

/F

Specifies that MS OS/2 *MULTIUSER* should not discard the extended attributes of a file if the *destination* file system does not support extended attributes. In this case, the system does not copy the file.

Remarks

When you use the COPY command to append files to an existing file without changing the existing *filename* (including its extension), you must specify that *filename* as the first *source* file, followed by the names of the files you want to append.

When you copy a single file to a new file, the new file has the same date and time as the original. When you combine files into a new file, the new file has the current date and time.

When you use the COPY command to make a copy of an existing file, the system copies the extended attributes of that file to the new file. When you use COPY to append files to an existing file, the extended attributes of the existing file do not change. When you append files to a new file, the new file uses the extended attributes of the first of the appended files.

If you omit the /F option when you copy a file with extended attributes to a file system that does not support extended attributes, MS OS/2 *MULTIUSER* copies the file and discards the extended attributes. If the file requires the extended attributes, the system does not copy the file but displays an explanatory message on the screen.

To copy files that have long *filenames* from an HPFS disk or partition to a FAT disk or partition, you must give the files new names in the FAT filename format (*XXXXXXXX.XXX*).

You must have Read (R) permission on the *source* file and Create (C) on the *destination* file.

Examples

To copy the file *COMPANY.NEW* in the current directory on the current drive to the CUSTOMER directory on the disk in drive B, type the following:

```
COPY COMPANY.NEW B:\CUSTOMER
```

To copy all the files in the directory REPORTS on your fixed disk (drive C) to the disk in drive A, type the following:

```
COPY C:\REPORTS\* A:
```

To combine the files *INTRO.RPT*, *BODY.RPT*, and *SUM.RPT* from the current drive and directory and place them in a file called *REPORT* on the disk in drive B, type the following:

```
COPY INTRO.RPT + BODY.RPT + SUM.RPT B:REPORT
```

If you omit the *destination filename*, the system combines the files and stores them under the name of the first file specified.

To combine all files from the current drive and directory that have the extension *.TXT* into one file named *COMBIN.DOC* in the same directory, type the following:

```
COPY *.TXT COMBIN.DOC
```

To combine each file from the current drive and directory that has the extension *.TXT* with the corresponding file that has the extension *.REF* and place the results into files with the extension *.DOC* (for example, *VIDEO.TXT* and *VIDEO.REF* would be combined as *VIDEO.DOC*), type the following:

```
COPY *.TXT + *.REF *.DOC
```

To copy what you type at the keyboard (CON) to the file *TEXT.TXT*, type the following:

COPY CON TEXT.TXT

After typing your text, press **CTRL+Z** and then press **ENTER** to complete the command.

CREATEDD (Create Dump Diskette)

Description

Prepares a disk that can be used to copy the contents of memory.

Syntax

CREATEDD *drive*:

Parameter

drive:
Specifies the drive that contains the disk to be used for copying memory contents.

Remarks

A copy of memory contents may be helpful in identifying system problems. To copy memory contents to a disk, insert the disk prepared by using CREATEDD into the disk drive and press **CTRL+ALT+NUMLOCK** twice on the system console. If there is more information in memory than can fit on one disk, the system prompts you to insert additional disks. These additional disks need only to have been formatted. After a disk is full, you receive summary information for that disk and the system prompts you to insert another disk. When the copying operation is completed, the system prompts you to reinsert the disk created by using CREATEDD. The system then stops and you must restart it.

If REBOOT=OFF in *CONFIG.SYS*, **CTRL+ALT+NUMLOCK** twice is disabled on the system console.

Do not start the copying operation while the fixed disk is being read from or written to. This could disrupt directory information on the fixed disk, causing files to be lost.

Example

To prepare a disk in drive A for copying memory contents, type the following:

```
CREATEDD A:
```

DATE (Set or Display Date)

Description

Queries and sets the system calendar by specifying the current date.

Syntax

DATE [*month-day-year*]

Parameters

month
Specifies a number in the range 1 through 12.

day
Specifies a number in the range 1 through 31.

year
Specifies a four-digit number in the range 1980 through 2079. You can abbreviate this number by using the last two digits of the year, in the range 80 through 79.

Remarks

You can use slashes (/) or periods (.) instead of hyphens (-) to separate the month, day, and year.

If you type DATE by itself, MS OS/2 *MULTIUSER* displays the current date and then prompts you for a new date. If you don't want to change the date, just press **ENTER**.

If you used the COUNTRY configuration command in your *CONFIG.SYS* file to alter the date format (for instance, to specify a date as *day-month-year*), the DATE command reflects that change.

Security Class Restrictions

Administrator and Operator classes have permission to change the date. User and Guest classes can query the date. An error message will be displayed if a user who is not authorized to do so attempts to change the date.

Example

To set the system date to May 1, 2017, type the following:

```
DATE 5-1-17
```

DDINSTAL (Install Device Driver)

Description

Provides an automated way to install new devices and their device drivers on your system without running the MS OS/2 *MULTIUSER* installation program again. The DDINSTAL utility copies the device driver to your system and adds the appropriate device command to your *CONFIG.SYS* file.

Syntax

```
DDINSTAL
```

Remarks

To install a device by using the DDINSTAL utility, type DDINSTAL and press **ENTER**. The utility prompts you to insert the disk that contains the device driver to be installed. Then simply follow the instructions on the screen to complete the installation.

DDINSTAL will fail if the install description file (*.DDP*) attempts to install a change to *OS2.INI*.

For more information and restrictions on DDINSTAL, see Chapter 10, "Using Citrix *MULTIUSER* Device Drivers" in the *Citrix MULTIUSER System Administrator's Guide.*

Security Class Restrictions

This utility can be executed only by the Administrator or Operator class.

DEL (Delete Files)

Description

Deletes a file or group of files. You can type ERASE instead of DEL if you prefer; they both have the same effect.

Syntax

DEL [*drive*:][*path*]*filename* [...] [/P]

Parameter

filename

Specifies the name of the file to be deleted. You can use wildcard characters to delete more than one file in a directory; you can also specify more than one file by typing the *filenames* individually, separated by spaces.

Option

/P

Asks you for confirmation before deleting each file.

Remarks

You can type DEL * to delete all the files in the current directory. (You cannot, however, delete the directory itself by using DEL.) To delete all the files in another directory, type only DEL followed by the directory name. To prevent you from accidentally deleting important files, the system displays a message like the following when you type either of the preceding commands:

```
C:*, Are you sure (Y/N)?
```

You must have Delete (D) permission to delete a file.

WARNING: Once you have deleted a file from your disk, it is gone; you cannot recover it. Be sure you have specified the right file or files, with the correct path, before you press **ENTER**.

If you are deleting a group of files and the system cannot delete one of them (for example, if one of the files you specified does not exist), it deletes the rest of the files in the group and then displays an error message identifying the file it could not delete and the reason.

DETACH (Start Non-interactive Program)

Description

Detaches a process to run in the background while you go to another task.

Syntax

DETACH *command* [*options*]

Parameters

command
Specifies any MS OS/2 *MULTIUSER* program or command that does not require you to type input from the keyboard.

options
Specifies any valid options that the *command* can accept in the command line.

Remarks

When you detach a process, the system starts it as an independent process, displays the process identification number (*processID*), and immediately displays the command prompt. You can type other commands while the detached process is running in the background.

You should not detach programs that require keyboard input.

You can run programs in the background sequentially by listing their names in order, separated by ampersands (&).

You cannot stop a detached process; it must complete itself on its own. If you delete the parent process (quit the command interpreter, for instance), the detached process still runs until it is finished. However, if the parent process is the only (or last) foreground program in the login and you end it, the login is terminated (logged out). Any process that you detached will then also be terminated.

If you try to DETACH a program that should not be run in the background, you could ruin files or lose valuable information. The documentation for the program should tell you whether it can be detached safely.

Examples

To create an alphabetically sorted listing of the *SORT.IN* file and put it in a file called *SORT.OUT*, and to have this process run in the background so that you can run another process while it is sorting, type the following:

```
DETACH SORT < SORT.IN > SORT.OUT
```

To copy all the files from the current directory on the current drive to drive D and then place the directory listing in *FILES.LST*, and to have these processes run in the background, type the following:

```
DETACH COPY * D: & DIR D: > FILES.LST
```

DIR (Display Directory Contents)

Description

Displays a list of the files on a disk or in a directory, with information about the size of each file and when it was created, the number of files in the directory, and the number of bytes free on the disk.

Syntax

DIR [*drive*:][*path*][*filename*] [...] [/P] [/W | /F] [/N]

Parameters

path

Specifies the directory whose listing you want to display.

filename

Specifies a particular file whose listing you want to display. If you specify a *filename*, the system displays information about that file alone. You can use wildcard characters to get information about groups of files with similar names — for example, to compare the dates and sizes of several files with the same extension.

Options

/P

Displays the listing one screenful at a time.

/W

Lists only the *filenames* (including their extensions) and displays them across the width of the screen in several columns. If you are using a file system that recognizes long *filenames*, the width of the columns is determined by the length of the longest *filename*. This option cannot be used with the /F option.

/F

Lists the drive, path, and *filename* (including extension) for each file specified, but does not display any other information about the file. This option cannot be used with the /W option.

/N

(FAT file system only.) Displays the listing in the format used in HPFS, with the *filename* in the right-hand column and an additional column showing the size of each file's extended attributes.

Remarks

You can specify several drives, paths, or *filenames.* For each different drive or directory you specify, MS OS/2 *MULTIUSER* displays a separate list of files. If, however, you specify more than one file or group of files in the same directory, the system displays them in a single list for that directory.

If you type DIR by itself, the system displays information about all the files you have Read (R) or Execute (X) permission for in the current directory on the current drive. Files you do not have Read (R) or Execute (X) permission for are not shown.

If you are displaying directory listings for several directories or disks (or for several files specified individually) and the system cannot display a listing for one of them (for example, if one of the directories does not exist), it displays an error message for that file, directory, or disk, and continues displaying the rest of the listings.

DISCONN (Disconnect User Login)

Description

Disconnects a login from a terminal.

Syntax

DISCONN [*loginID*] [/PASSWORD:*password*] [/V] [/?]

DISCONN [\ *username.groupname*\ *loginID*] [/PASSWORD:*password*] [/V] [/?]

DISCONN [\ *terminalname*] [/PASSWORD:*password*] [/V] [/?]

Parameters

loginID
Identifies a login.

\ *username.groupname*\ *loginID*
Identifies a specific login belonging to the specified *loginname.*

\ *terminalname*
Identifies all logins connected to *terminalname.*

Options

/PASSWORD:*password*
Specifies the password associated with the *loginname* that owns the specified *loginID.* Only needed in cases where the specified *loginID* belongs to a *loginname* that is different from the current *loginname.*

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

If no *loginID* or *terminalname* is specified, DISCONN uses the current *loginID*. The *terminalname* must identify an active terminal. Wildcards may be used in the input parameter specifying the *loginID*. If a single wildcard is present (*), it is interpreted as a wildcard for the *loginID* and specifies all *loginIDs* in the system.

If the /PASSWORD option is not specified and the target *loginID* belongs to a *loginname* that is different from the current *loginname*, DISCONN will prompt for the password.

For a discussion on disconnected logins, refer to Chapter 5, section "Using Disconnected Logins" in the *Citrix MULTIUSER User's Guide.*

Security Class Restrictions

DISCONN may only be used to disconnect logins belonging to the current *loginname*, unless the current *loginname* has sufficient security classification to disconnect logins belonging to other *loginnames*. Administrator class and Operator class have full access to all DISCONN functions. User class and Guest class may use only DISCONN functions on logins that belong to the current *loginname*.

Examples

example 1: no options
DISCONN disconnects the current login.

```
DISCONN
```

example 2: *username*

DISCONN disconnects all logins belonging to *loginnames* with the *username* BRAD and all *groupnames* associated with the *username* BRAD.

```
DISCONN \BRAD
```

example 3: *loginname*

DISCONN disconnects all logins with the *loginname* BRAD.WRKGRP.

```
DISCONN \BRAD.WRKGRP
```

example 4: loginID

DISCONN disconnects the login with *loginID* of 12.

```
DISCONN 12
```

example 5: fully specified loginID

DISCONN disconnects the login with *loginID* of 10, belonging to the *loginname* of BRAD.WRKGRP.

```
DISCONN \BRAD.WRKGRP\10
```

example 6: *terminalname*

DISCONN disconnects all logins at the terminal TERM04.

```
DISCONN \TERM04
```

example 7: wildcards

DISCONN disconnects all logins in the system (if the user has the authority to do so).

```
DISCONN *
```

DISKCOMP (Compare Diskettes)

Description

Compares two floppy disks track by track.

Syntax

DISKCOMP [*drive1*:] [*drive2*:]

Parameters

drive1:
: Specifies the drive letter of the first disk being compared.

drive2:
: Specifies the drive letter of the second disk being compared.

Remarks

Since the DISKCOMP utility automatically determines the number of sides and sectors per track by looking at the format of the first disk, both disks must be of the same type (for instance, high-density 5¼-inch disks).

If you specify only one drive, DISKCOMP compares the floppy disk in *drive1* with the floppy disk in the current drive. If you specify the same drive for both *drive1* and *drive2*, DISKCOMP uses only that drive and prompts you to change disks as needed during the comparison. If you type DISKCOMP by itself, the utility uses only the current drive and prompts you to insert the two disks, as appropriate. (If the current drive is not a floppy-disk drive, you see an error message.)

You may have to RESERVE both diskette drives using the RESERVE command if the System Administrator has set permissions requiring this. Drives A and B normally will require RESERVE.

Example

To compare two high-density floppy disks when you have only one high-density disk drive (drive A), type the following:

```
DISKCOMP A: A:
```

The DISKCOMP utility prompts you to insert each disk in turn, as needed, during the comparison.

DISKCOPY (Copy Diskette Image)

Description

Makes a duplicate of a floppy disk.

Syntax

DISKCOPY [*drive1*:] [*drive2*:]

Parameters

drive1:
  Specifies the drive that contains the floppy disk to be copied (the *source* disk).

drive2:
  Specifies the drive that contains the floppy disk that will become the duplicate (the *destination* disk).

Remarks

The two disks must be of the same type (for example, high-density 5¼-inch floppy disks). If the *destination* disk is unformatted, DISKCOPY formats it with the same number of sides and sectors per track as the *source* disk.

You can copy a disk using only one drive, either by not specifying any drives or by specifying the same drive for both *source* and *destination*. DISKCOPY prompts you to insert the two disks as needed. If you type DISKCOPY by itself, the utility uses the current drive.

DISKCOPY writes over the information on the *destination* disk even if it does not have to format the disk, so any information that is already on the disk is lost.

You may have to RESERVE both diskette drives using the RESERVE command if the System Administrator has set permissions requiring this. Drives A and B normally require RESERVE.

Example

To copy the floppy disk in drive A to a floppy disk in drive B, type the following:

```
DISKCOPY A: B:
```

DPATH (Set Search Path for Data Files)

Description

Specifies which directories besides the current directory an application should search in order to find data files (files with extensions other than *.EXE*, *.COM*, *.BAT*, or *.CMD*).

Syntax

DPATH [; | [*drive*:]*path*[;...]]

Parameters

;

When used alone, (DPATH ;) clears all data-path settings. Semicolons are also used to separate multiple data paths.

path

Specifies the *path* of the directory that you want the application to search. You can specify more than one path, separating them with semicolons (;).

Remarks

If you type DPATH by itself, MS OS/2 *MULTIUSER* displays the current data path.

Each time you use the DPATH command, the new data path you specify takes the place of the previous path. The data path is stored in the system environment. Only applications that are written to take advantage of DPATH will use the data path specified.

The DPATH command affects only the current session. To specify a data path for all sessions, you must set the DPATH environment variable in your *CONFIG.SYS* file. For information about how to do this, see the SET command.

If you start a new command interpreter from within a session where DPATH is defined, the new session inherits the DPATH setting. You can change this setting in the new session without affecting the settings in the parent session.

The DPATH command works much the way the PATH command does, except that DPATH is used by an application to search for data files, whereas PATH is used by commands and utilities outside of applications.

If your access is denied for a path in DPATH, it is skipped.

Example

To instruct an application to search for data files in the SYMPHONY directory and the COUNTRY subdirectory of the BANDS directory on the current drive, and in the ROCK subdirectory of the BANDS directory on drive D, type the following (all on a single line):

```
DPATH \SYMPHONY;\BANDS\COUNTRY;
D:\BANDS\ROCK
```

E (System Editor)

Description

Issuing E starts the full screen System Editor. System Editor allows you to display and modify data files. A complete description of System Editor is found in Chapter 7, "Using Citrix *MULTIUSER* System Editor" of the *Citrix MULTIUSER User's Guide.*

Syntax

E [*filename*] [*filename*] ...

Parameter

filename
Specifies the name of the file you wish to edit. Multiple *filenames* can be listed. The *filename* parameter is optional.

Remarks

If no *filenames* are entered, a blank screen will be displayed. System Editor commands are available to specify files to edit while in the full screen mode.

You must have Read (R) permission to display the file and Write (W) permission to update the file. If the file is new, you need Create (C) permission in the directory in which you wish to place the file.

This utility preserves extended attributes and security attributes.

Example

E MYFILE.DAT
Displays the file MYFILE.DAT in full screen mode and allows editor commands to be issued to change the file.

EAUTIL (Save and Restore Extended File Attributes)

Description

Deletes or copies (splits) the extended attributes from a data file or directory and stores them in a separate file, or copies (joins) the extended attributes back to the original data file or directory. The EAUTIL utility enables applications that do not recognize extended attributes to manipulate files and directories without losing their extended attributes. The utility also enables you to send files over a network or copy them to systems that do not recognize extended attributes without losing their extended attributes.

Syntax

EAUTIL [/S [/R] [/P] | /J [/O | /M] [/P]] *filename* [*holdfile*]

Parameters

filename

Specifies the data file or directory to copy the extended attributes from or back to.

holdfile

Specifies the file to store the extended attributes in or copy them back from. If you do not specify a filename for *holdfile* when splitting extended attributes, the EAUTIL utility creates a file with the same name as *filename* and stores it in a directory (as a subdirectory of the current directory) named EAS. If you do not specify a filename for *holdfile* when joining extended attributes, EAUTIL looks in the EAS directory as a subdirectory of the current directory for a file with the same name as *filename.*

Options

When you use the EAUTIL utility to split the extended attributes from a file or directory, you use the /S option. When you specify /S, you can also specify the /R and /P options.

/S

Copies the extended attributes from *filename* to *holdfile.* This is the default option.

/R

Replaces the contents of *holdfile* with the extended attributes currently attached to *filename.* This option is used with the /S option. If *holdfile* already exists, you must specify the /R option in order to copy the extended attributes from *filename*; otherwise, the operation will fail.

/P

When used with the /S option, preserves the extended attributes attached to *filename*. If this option is not set, the EAUTIL utility deletes the extended attributes from *filename* after copying them to *holdfile*.

When you use the EAUTIL utility to join extended attributes to a file or directory, you use the /J option. When you specify /J, you can also specify the /O or /M options and the /P option. If you specify the /J option and extended attributes are already attached to *filename*, you must specify either the /O or /M option or the operation will fail.

/J

Copies the extended attributes from *holdfile* back to *filename*.

/O

Overwrites the extended attributes attached to *filename*, replacing them with the extended attributes in *holdfile*. This option is used with the /J option. You cannot specify the /O option with the /M option.

/M

Adds (merges) the extended attributes in *holdfile* to those in *filename*. This option is used with the /J option. You cannot specify the /M option with the /O option.

/P

When used with the /J option, preserves *holdfile* after the extended attributes in it have been copied to *filename*. If the /P option is not set, the EAUTIL utility deletes *holdfile* after joining the extended attributes to *filename*.

NOTE: If your system crashes while you are running the EAUTIL utility, the splitting or joining operation may be incomplete. If you are splitting extended attributes from a file or directory, you may have complete extended attributes in one location and incomplete ones in the other, but you won't know whether it is *filename* or *holdfile* that contains the complete set. To find out, run EAUTIL again, specifying a different name for *holdfile*; then compare the sizes of the two *holdfiles*. The larger file contains the complete extended attributes and you can delete the smaller file.

If you are joining extended attributes to a file or directory when the system crashes, the extended attributes in *filename* may be incomplete. To remedy this, simply run EAUTIL again and use the /O (overwrite) option (even if you are using the /M option when the system crashes).

You need Read (R) permission to copy extended attributes (EAs) and Attribute (A) permission to put them back in the file.

Examples

To delete the extended attributes from the file *REPORT* and store them in a new file called *REPORTEA* so that you can safely copy both files to a system running MS DOS or a version of MS OS/2 earlier than 1.2 without losing the extended attributes, type the following:

```
EAUTIL /S REPORT REPORTEA
```

To copy the extended attributes from the file *REPORTEA* back to the file *REPORT* after you have copied both files to a system running MS OS/2 version 1.2 or MS OS/2 *MULTIUSER*, type the following:

```
EAUTIL /J REPORT REPORTEA
```

ECHO (Echo Batch Commands)

Description

Turns ON or OFF the feature that displays batch file commands on the screen while they are being processed or simply displays the specified message on the screen. Although the ECHO command can be used from the command prompt, it is designed to be placed in a batch file.

Syntax

ECHO [ON | OFF | *message*]

Parameter

message
Specifies a line of text to be displayed.

Remarks

If you type ECHO by itself, the system displays the current setting.

Ordinarily, the system displays (echoes) commands in a batch file on the screen when it receives them. You can turn off this feature by specifying ECHO OFF in the batch file; none of the subsequent lines appear on the screen until the system encounters an ECHO ON command. You can also turn off the echoing of any individual command in a batch file, including the ECHO OFF command, by preceding the command with the @ symbol.

You can display a message from a batch file (whether the ECHO feature is turned ON or OFF) by specifying an echo message in the file. If you want to display a message of more than one line, you must start each line with ECHO.

Example

To turn the ECHO feature OFF and then have a batch file display the message "This batch file formats and checks new disks," type the following in the batch file:

```
@ECHO OFF
ECHO This batch file formats and checks
ECHO new disks.
```

ENDLOCAL (Restore Environment)

Description

Restores the drive, directory, and environment settings that were in effect before the SETLOCAL command changed them. Use this command only in a batch file.

Syntax

```
ENDLOCAL
```

Remarks

You can use multiple SETLOCAL commands in a batch file without including corresponding ENDLOCAL commands; each succeeding SETLOCAL overrides the previous one, in effect acting as an ENDLOCAL command for the previous SETLOCAL. If there is no ENDLOCAL command after the last SETLOCAL command in a batch file, the system restores the original drive, directory, and environment settings when the batch file ends.

Example

To use SETLOCAL in a batch file to set an alternate search path for certain commands to use and to reset the original path after those commands are finished, type the following lines in the batch file:

```
SETLOCAL
PATH C:\TEST;A:\TEMP
  .
  .
  .
ENDLOCAL
```

ERASE (Delete Files)

See DEL (Delete Files).

EVENTS (Manage Event Log)

Description

This utility is used to perform several Event Management operations on the System Event Log data. It is used to format log data, such as the Security Audit Trail. It is also used to force the online log file to switch to another file.

Syntax

EVENTS [*filename*] [/MON:*sec*] [/AUDIT]
[/CAT:*eventletters*] [/DATE:*mmdd*] [/STYLE:*s*]
[/USER:*username*] [/GROUP:*groupname*]
[/TERM:*terminalname*] [/G+:*accessletter*]
[/D+:*accessletter*] [/MASK:*hexmask*] [/EID:#] [/?]

EVENTS [/SWAP]

Parameter

filename

Specifies the name of the log file to be formatted. If this parameter is not entered, the current active log file is used. If a path is not specified the current path is searched first, and if the file is not found in the current path, the system log path is searched.

Options

/MON:*sec*

Specifies that the active log should be monitored for new log data. The *sec* parameter indicates the number of seconds to wait prior to checking for new data in the logfile. If this parameter is specified, *filename* cannot be specified.

/CAT:*eventletters*

Specifies the event category selection list. The default is all categories are selected.

/AUDIT

Shorthand to specify that all audit events should be selected. Other selection data (such as, /CAT, /EID, /MASK) pertaining to the event will be ignored and should not be specified.

/DATE:*mmdd*

Specifies the month and day to be selected. Only records matching this month and day are formatted.

/STYLE:*s*

Specifies the formatting style to use. This can be D for "detailed," C for "compressed," or N for "normal." N is the default.

/USER:*username*

Specifies the *username* to be selected. Only records generated by the given *username* are formatted. A serial number can be used in place of *username* by preceding the decimal number with a pound sign (#).

/GROUP:*groupname*

Specifies the *groupname* to be selected. Only records generated by the given *groupname* are formatted. A serial number can be used in place of *groupname* by preceding the decimal number with a pound sign (#).

/TERM:*terminalname*

The *terminalname* is the name given to the terminal by the System Administrator when the terminal was configured. QUERY TERMINAL can be used to determine the *terminalname.*

Only records generated with the given *terminalname* are formatted. A serial number can be used in place of *terminalname* by preceding the decimal number with a pound sign (#).

/G+:*accessletter*

If this switch is present, access granted audit events are selected for formatting. The specific events requested are listed as *accessletter.* An asterisk (*) for *accessletter* selects all logged audit events for which access is granted. See CHANGE AUDIT for a description of *accessletter.*

/D+:*accessletter*

If this switch is present, access denied audit events are selected for formatting. The specific events requested are listed as *accessletter.* An asterisk (*) for *accessletter* selects all logged audit events for which access is denied. See CHANGE AUDIT for a description of *accessletter.*

/MASK:#

Specifies hex bit mask to be used in selecting an event ID for formatting. The default is all events are selected within the category. This can be specified in hex format as 0x#.

/EID:#

Specifies a specific event ID to select for formatting. The default is all events are selected within the category.

/SWAP

This switch indicates that the system should SWAP to another log file. No other parameters are valid with this parameter. See Remarks.

eventletters

A list of letters, each of which denotes a specific system event category. These letters are:

A	Security access audit
E	Errors
M	Multiuser events
P	Profile update events
R	Resource use events
S	Security secondary audit events
T	Terminal subsystem events

/? (help)

Displays the syntax for the utility and information about the utility's options.

Remarks

When log formatting is selected, the output is directed to standard out, which is normally the display. You can file this formatted data by redirecting it to a file.

Formatting does not remove data from the input file, so the same data can be reviewed in different ways by reformatting.

The active system log is:

\OS2\SYSTEM\LOG0001.DAT

When the /SWAP option is selected, the current log is closed and renamed and a new LOG0001.DAT file is created and opened for more logging. Closed logs are renamed as follows:

\OS2\SYSTEM\LOG*mmdd.###*

where *mmdd* is the month and day, and *###* is a decimal serial number.

You should perform a SWAP on a regular basis so that the active log does not fill up. See Chapter 14, "System Event Logging," in the *Citrix MULTIUSER System Administrator's Guide*. The closed logs can then be formatted or copied to diskette to free up disk space. The SWAP process shuts down logging temporarily so it is best to do this while the system is relatively idle.

Security Class Restrictions

This utility is available only to an Operator or Administrator.

Examples

EVENTS LOG0801.001 >LOG0801.OUT
Formats the first log created on 8/1 and puts the output in a file.

EVENTS /AUDIT
Formats a report of all of the audit events in the active log.

EVENTS /SWAP

Causes the system to SWAP to a new log file.

EXIT (End Command Interpreter)

Description

Ends the current command interpreter and returns control to the program from which it was started.

Syntax

```
EXIT
```

Remarks

If you have used CMD to start another MS OS/2 *MULTIUSER* command interpreter, the EXIT command ends that command interpreter and returns you to the parent command interpreter. If you type EXIT from the parent command interpreter, the session ends.

NOTE: If your first program is *CMD.EXE* (instead of the Program Selector), typing EXIT from your last session is like a logout.

EXTPROC (External Batch Processor)

Description

Defines an external batch processor for a batch file. Use this command only in a batch file.

Syntax

EXTPROC [*drive*:][*path*]*filename* [*options*]

Parameters

filename

Specifies the name of the file that contains the external batch processor. The *filename* must include the extension. You can also include a *drive* and a *path*, if necessary.

options

Specifies any valid options for the new batch processor.

Remarks

By putting this command as the first line of your batch file, you cause the system to start a different batch processor and use that to run the batch file.

Example

To run a batch file by using a batch processor called *BORNE.EXE*, which is in the BATCH directory on your fixed disk (drive C), instead of using CMD, type the following as the first line of your batch file:

```
EXTPROC C:\BATCH\BORNE.EXE
```

FDISK (Disk Partitioning)

Description

Creates or deletes the primary MS OS/2 *MULTIUSER* partition, the extended partition, and any logical drives from the fixed disk where you start your system. Prepares a fixed disk for formatting.

Syntax

```
FDISK [/D]
```

Option

/D

This switch deletes all MS OS/2 *MULTIUSER* compatible partitions and logical drives on the boot disk. You must run this option from the Install diskette.

Remarks

To create or delete partitions without deleting the primary partition on your start-up drive, use FDISK at the command prompt.

The FDISK command displays a series of menus to help you partition your fixed disk for MS OS/2 *MULTIUSER*. With FDISK, you can do the following:

- Create a primary partition, an extended partition, or a logical drive in the extended partition.
- Change the active partition.
- Delete a primary partition, an extended partition, an enhanced partition, or a logical drive (from the extended partition).
- Display partition data or information about a logical drive in the extended partition.
- Select the next fixed disk for partitioning on a system with multiple fixed disks.

Partitions can only be modified in maintenance mode. For more information on how to use FDISK, see Chapter 3 of the *Citrix MULTIUSER System Administrator's Guide.*

Security Class Restrictions

Only the Administrator or Operator class has permission to use the FDISK utility.

FIND (Search File for Text String)

Description

Searches the file or files you specify or the input you give it for a specific string of text and displays all the instances of that string that it finds.

Syntax

FIND [/V] [/C] [/N] "*string*" [[*drive*:][*path*][*filename*] [...]]

Options

/V
: Displays all lines that do not contain *string*.

/C
: Displays only the total number of lines found that contain *string*.

/N
: Displays each line that contains *string*, with a number in front of it that indicates its position within the file.

Parameters

string
: Specifies the group of alphanumeric characters you want to search for. You must enclose the string in double quotation marks (" "). Since the FIND utility is case sensitive, you must type uppercase and lowercase letters exactly as you want the utility to search for them. If you are searching for a *string* that contains double quotation marks, you must use two double quotation marks for each double quotation mark that is part of the *string*.

filename
: Specifies the file in which to search for *string*. If the file is not in the current directory of the current drive, you must also specify a *drive* and/or *path*. You cannot use

wildcard characters when specifying *filenames*, but you can specify several files in one find operation. If you do not specify a *filename*, FIND searches standard input.

Remarks

If you specify /C with /V, the FIND utility displays the number of lines that do not contain *string*. If you specify /C with /N, FIND ignores /N.

You must have Read (R) permission for the files.

Example

To search for the *string* "Hello" in the files *CHAPTER1.DOC* and *CHAPTER2.DOC* on drive A and to display the lines that contain the *string* along with their relative line numbers, type the following on one line:

```
FIND /N "Hello" A:\CHAPTER1.DOC A:\CHAPTER2.DOC
```

FOR (Repeat Command)

Description

Performs a command for a set of files or other items that you specify. Although the FOR command can be used from the command prompt, it is designed to be placed in a batch file.

Syntax

FOR [%]%*x* IN (*item* [...]) DO *command*

Parameters

%%*x*

Specifies the variable that will be affected by command. The values of *item* are substituted sequentially for this variable. You can use any single letter or number for *x*.

item

Specifies a file or other *item* you want to substitute for %%*x* so that command will affect it. You can specify multiple *items*, separated by spaces. These *items* are substituted sequentially for %%*x*. You can use wildcard characters in *item* name, and you can use replaceable parameters as *items*.

command

Specifies the *command* you want to perform on the items you have specified. This argument can be any MS OS/2 *MULTIUSER* command or utility. You can also include any valid arguments for the command or utility that you specify.

Remarks

The FOR command substitutes the first item for the placeholder %%*x* and performs a command on that item; then FOR substitutes the second item for %%*x* and performs the command again and so on until no items remain. If you use wildcard characters in an *item* name, the %%*x* is replaced once for each matching *filename*. For more information about replaceable parameters, see Chapter 6 of the *Citrix MULTIUSER User's Guide.*

If you use FOR directly from the command line, use only one percent sign in front of *x* (%*x*). In a batch file, however, you must use two percent signs to distinguish this FOR variable from a replaceable parameter, which can be a number (%0-%9) or a string (%variable%).

Example

To delete in turn each of the three files *REPORT*, *MEMO*, and *ADDRESS*, type the following in a batch file:

```
FOR %%F IN (REPORT MEMO ADDRESS) DO DEL %%F
```

FORMAT (Initialize Disk or Diskette)

Description

Prepares a disk or partition so that it can store files.

Syntax

FORMAT *drive*: [/4] [/T:*tracks*] [/N:*sectors*] [/V:*label*] [/F:*size*] [/FS:*filesystem*]

Parameter

drive:
Specifies the *drive* that contains the disk you want to format. This information is required.

Options

/4
(FAT file system only) Formats a 5¼-inch, double-sided, low-density disk in a high-density drive. If you are using a low-density drive, you may not be able to reliably read disks formatted with this option.

/T:*tracks*
(FAT file system only.) Formats a 3½-inch disk to the number of tracks specified by *tracks*. The size of the specified drive determines the default value for *tracks*.

/N:*sectors*
(FAT file system only.) Formats a 3½-inch disk to the number of sectors specified by *sectors*. The size of the specified drive determines the default value for *sectors*.

/V:*label*

Specifies the volume *label* — a name used by programs to identify the disk. The *label* can be up to 11 characters. If you do not specify this option, the FORMAT utility prompts you for a volume *label* after formatting is complete. If the *label* includes blank spaces, you must enclose the *label* in double quotation marks.

/F:*size*

(FAT file system only) Specifies the memory capacity of a disk in kilobytes (or for a 1.2-megabyte or 1.44-megabyte disk, in megabytes). You can use this option instead of the /4, /T, and /N options to simplify the process of formatting a floppy disk. The easiest way to specify *size* is as the number of kilobytes of memory on the disk (360, 720, 1200, or 1440), but you can add an abbreviation (for example, 360K, 360KB, 1.2M, 1.2MB) if you prefer.

/FS:*filesystem*

Specifies the *filesystem* for which to format the disk or partition. This can be any *filesystem* supported by MS OS/2 *MULTIUSER*. To format a disk or partition for the High-Performance File System (HPFS), specify HPFS as the *filesystem* argument. When you use the /FS:HPFS option, the FORMAT utility adds the /AUTOCHECK:*drive* option to the end of the IFS=C:\OS2\HPFS.IFS /C:64 line in your *CONFIG.SYS* file. Whenever you subsequently restart your system, the /AUTOCHECK option causes the system to run CHKDSK /F for any HPFS drives that were shut down improperly.

Remarks

You must use the FORMAT utility to format all new disks so that MS OS/2 *MULTIUSER* can use them. The FORMAT utility creates the directory and the *filesystem* tables on the disk and assigns the disk a unique serial number. FORMAT uses the drive type to determine the default format for the disk.

You cannot format a disk that is in use, nor can you format your start-up disk or the disk that you started the FORMAT utility from.

You cannot format a floppy disk for HPFS.

WARNING: The FORMAT utility erases all information that is already on the disk, so be sure you specify the correct drive.

Security Class Restrictions

Only Administrator and Operator classes can FORMAT a fixed disk partition. Any user can execute FORMAT for a diskette. An error message is displayed if a user who is not authorized to do so attempts to use the FORMAT utility.

Example

To format a floppy disk in drive A and give it the label OLD LETTERS, type the following:

```
FORMAT A: /V:"OLD LETTERS"
```

GOTO (Transfer Control in Batch File)

Description

Directs the system to go to a particular line in a batch file and continue processing commands from that point. Use this command only in a batch file.

Syntax

GOTO *label*

Parameter

label

Specifies the location in the batch file where MS OS/2 *MULTIUSER* should continue processing commands. *label* can be any string you choose, but the GOTO command uses only the first eight valid characters to identify the *label.* Spaces, tabs, and certain common separators such as the equal sign (=) and the semicolon (;) are not valid characters in a *label*; it is best to use only letters and numbers.

Remarks

The specified *label* should appear on a line by itself, preceded by a colon (:); this line is ignored in batch processing except as a marker for the GOTO command. The *label* line can be either before or after the GOTO command in the file.

The GOTO command simply ignores invalid separators if they appear in the *label* line but when it encounters a space or a tab, it stops reading the *label.* The following *label* lines, then, are equivalent:

```
:LAB
:LAB EL
:--LAB EL
:==LAB
==:;;LAB EL
```

Example

To format the disk in drive A and either go to the end of the file if no errors occur or display an error message if the formatting isn't completed successfully, type the following in a batch file:

```
@ECHO OFF
FORMAT A:
IF NOT ERRORLEVEL 1 GOTO END
ECHO An error occurred during formatting.
:END
ECHO End of batch file.
```

HELP (Get Help Message)

Description

Displays help information about MS OS/2 *MULTIUSER* and about error or warning messages displayed in a session.

Syntax

HELP [ON | OFF | *messageID*]

Parameters

ON

Displays a help line at the top of your screen. This line tells you how to get help and how to get back to the Program Selector. The HELP ON command replaces your current prompt setting with the help line and the current drive letter.

OFF

Removes the help line from your screen. This is the default setting.

messageID

Identifies the particular Help message that you want more information about. The message identification number consists of three letters — for example, SYS — followed by a four-digit number. For system messages, you can also specify the number without the letters and leading zeros (for example, SYS0002 can be specified as 2).

Remarks

If you type Help by itself, the utility displays a list of Help options and information.

For MS OS/2 *MULTIUSER* error messages, use this format:

HELP CTX*####*

where *####* is the MS OS/2 *MULTIUSER* error message number.

Example

To get more information about the error message "SYS0002: File not found", type the following:

HELP SYS0002

MS OS/2 *MULTIUSER* then displays the following information:

```
SYS0002: The system cannot find the file
specified.
EXPLANATION: The file named in the command does
not exist in the current directory or search path
specified. Or, the filename was entered
incorrectly.
ACTION: Retry the command using the correct
filename.
```

IF (Conditional Command)

Description

Directs MS OS/2 *MULTIUSER* to carry out the given *command* only if the specified condition is met. If you include the word NOT, MS OS/2 *MULTIUSER* carries out the command only if the condition is not met. Although the IF command can be used from the command prompt, it is designed to be placed in a batch file.

Syntax

IF [NOT] *condition command*

Parameters

condition
Specifies one of three conditions that determine whether MS OS/2 *MULTIUSER* carries out the *command*:

ERRORLEVEL *number*
When a program finishes, it sends an exit code to MS OS/2 *MULTIUSER*. If the exit code returned by the last program you ran was greater than or equal to *number*, MS OS/2 *MULTIUSER* carries out the *command.*

string1==string2
If the first *string* is exactly the same as the second *string*, MS OS/2 *MULTIUSER* carries out the *command.* Uppercase or lowercase is significant. The strings must not include separators such as commas, semicolons, equal signs, and spaces.

EXIST [*drive*:][*path*]*filename*

If the *filename* exists in the specified directory, MS OS/2 *MULTIUSER* carries out the *command*. You can specify a *drive* and a *path* before *filename*; otherwise, MS OS/2 *MULTIUSER* looks for *filename* in the current directory.

command

Specifies the *command* to carry out if the given condition is met.

Example

To display the message "Can't find data file" if the file *BOOK.DAT* does not exist in the current directory, type the following in a batch file:

```
IF NOT EXIST BOOK.DAT ECHO Can't find data file
```

KEYS (Retrieve Previous Commands)

Description

Turns ON or OFF the special functions of the **DIRECTION** keys and editing keys that enable you to edit commands on the command line. You can retrieve any command that is in the command-line buffer, edit it, and use it at the command prompt.

Syntax

```
KEYS [ON | OFF | LIST]
```

Parameters

ON

Turns on the special functions of the **DIRECTION** and editing keys used to edit commands on the command line and opens the command-line buffer so that each subsequent command is added to the buffer.

OFF

Turns off the special command-line editing functions of the **DIRECTION** and editing keys and closes the command-line buffer so that subsequent commands are not added to the buffer.

LIST

Lists the contents of the command-line buffer. The buffer contains a list of the last several commands you typed at the command prompt, numbered from oldest (1) to most recent.

Remarks

When you first use the KEYS command, MS OS/2 *MULTIUSER* sets an environment variable (KEYS=ON or KEYS=OFF); each subsequent time you type KEYS ON or KEYS OFF, the environment variable changes to reflect the new command. (No KEYS variable appears in the environment until the first time you use the KEYS command, unless you have previously set the variable in your *CONFIG.SYS* file.)

The KEYS command affects only the current MS OS/2 *MULTIUSER* session.

When KEYS is ON, the **DIRECTION** and editing keys have the following effects:

Key	Editing Function
ESC	Clears the current command line and returns the cursor to the position immediately following the prompt.
HOME	Returns the cursor to the position immediately following the prompt.
Key	Editing function.
END	Places the cursor in the position immediately following the last character you typed (even if that character is a space).
INS	Turns insert mode on and off. The cursor appears as a half box in insert mode. As you insert characters, existing characters to the right of the cursor move to the right. At the beginning of each new command line, insert mode is turned off.
DEL	Deletes the character marked by the cursor. As you delete characters, existing characters to the right of the cursor move to the left.
LEFT	Moves the cursor left one character.
RIGHT	Moves the cursor right one character.
UP	Moves to the previous command listed in memory and brings that command to the command line.
DOWN	Moves to the next command listed in memory and brings that command to the command line.

Key	Editing Function
BACKSPACE	Moves the cursor back (left) one character and deletes that character. Existing characters to the right of the cursor move to the left.
CTRL+LEFT	Moves the cursor left to the first character of the previous word.
CTRL+RIGHT	Moves the cursor right to the first character of the next word.
CTRL+END	Deletes all characters from the current cursor position to the end of the command line.
CTRL+HOME	Deletes all characters from the beginning of the command line up to (but not including) the current cursor position.
ENTER	Sends the information on the command line to the command interpreter, adds it to the list of commands in memory, and turns off insert mode.

KEYS ON remains in effect until you type KEYS OFF. MS OS/2 *MULTIUSER* adds commands to the command-line buffer only while the KEYS command is active, but it retains the buffer in memory until you end the session. Typing KEYS OFF does not empty the buffer; if you type KEYS ON again in the same session, MS OS/2 *MULTIUSER* adds any subsequent commands to the existing commands in the buffer.

The command-line buffer cannot be larger than 64K, but it might be smaller if there is not enough memory to extend the list. When the buffer is full, MS OS/2 *MULTIUSER* discards the oldest command to make room for the newest.

Only commands you type at the keyboard are added to the command-line buffer. Commands in a batch file or commands that are part of redirected input are not added. If you type KEYS by itself, MS OS/2 *MULTIUSER* displays the current setting.

KILL (Terminate Process(es))

Description

Terminates a process.

Syntax

KILL [*processID*] [/V] [/?]

KILL [\ *username.groupname*\ *loginID*\ *sessionID*\ *processID*]
[/V] [/?]

KILL [\ *terminalname*] [/V] [/?]

Parameters

processID
Identifies a process.

\ *username.groupname*\ *loginID*\ *sessionID*\ *processID*
Identifies a specific process belonging to the specific user context (loginname, login, session).

\ *terminalname*
Identifies all processes running at *terminalname* regardless of user context.

Options

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

If no *processID* or *terminalname* is specified, KILL displays an error message. If *terminalname* is specified, it must identify an active terminal.

Wildcards may be used in the input parameter specifying the *processID*. If a single wildcard is present (*), it is interpreted as a wildcard for the *processID* and specifies all *processIDs* in the system.

NOTE: A session will terminate when all processes running in it terminate. A login will terminate when all of its sessions have terminated. When a login terminates, it is "logged out". KILL may be used to terminate sessions or logins by terminating all processes in the sessions or logins.

Security Class Restrictions

KILL may only be used to terminate processes belonging to the current *loginname*, unless the current *loginname* has sufficient security classification to terminate processes belonging to other *loginnames*. Administrator class and Operator class have full access to all KILL functions. User class and Guest class may only use KILL functions on processes that belong to the current *loginname.*

Examples

example 1: *processID*

KILL terminates the specified process with *processID* of 8.

```
KILL 8
```

example 2: *sessionID*

KILL terminates all processes in session 3 of all logins in the system if the current *loginname* has the necessary security classification. This will terminate every session with *sessionID* 3.

```
KILL \.*\*\3
```

example 3: *loginID*

KILL terminates all processes in all sessions in the login with *loginID* 12.

```
KILL \*\12
```

example 4: *loginname*

KILL terminates all processes in all sessions in all logins under the *loginname* BRAD.WRKGRP.

```
KILL \BRAD.WRKGRP
```

example 5: *terminalname*

KILL terminates all processes at the terminal TERM04.

```
KILL \TERM04
```

LABEL (Disk Volume Label)

Description

Assigns a disk a new volume label.

Syntax

LABEL [*drive*:] [*label*]

Parameters

drive:

Specifies the *drive* that contains the disk to which you want to give a new volume label.

label

Specifies the new volume *label* for the disk in the specified drive. The *label* can be up to 11 characters, including spaces. Do not use any of the following characters in a volume *label*:

* ? / \ | . , ; : + = < > [ ] & ^ "

If you do not specify a drive, MS OS/2 *MULTIUSER* assigns *label* to the disk in the current drive.

Remarks

Programs use the volume *label*, which is simply a name for the disk, to identify the disk. The DIR command displays the volume *label* in addition to information about a disk's contents.

If you type LABEL by itself, the utility displays the volume *label* of the disk in the current drive and prompts you to either type a new *label* or press **ENTER** to retain the current label.

Security Class Restrictions

Any user can label a diskette; however, only Administrators and Operators can label a fixed disk partition.

Example

To give the volume label OLDSTUFF to the disk in drive A, type the following:

```
LABEL A:  OLDSTUFF
```

LOGOUT (End Login)

Description

Terminates the login.

Syntax

LOGOUT [*loginID*] [/V] [/?]

LOGOUT [\ *username.groupname*\ *loginID*] [/V] [/?]

LOGOUT [\ *terminalname*] [/V] [/?]

Parameters

loginID
Identifies a login.

\ *username.groupname*\ *loginID*
Identifies a specific login belonging to the specific *loginname.*

\ *terminalname*
Identifies all logins connected to *terminalname.*

Options

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

LOGOUT is not required to terminate the current login. When you exit the last session in the login, the login is terminated. It is recommended that you terminate your login by switching to each program and ending the program normally. This allows each program to terminate without losing any data.

If no *loginID* or *terminalname* is specified, LOGOUT terminates the current login. If the *terminalname* is specified, it must identify an active terminal. Wildcards may be used in the input parameter specifying the *loginID.* If a single wildcard is present (*), it is interpreted as a wildcard for the *loginID* and specifies all *loginID*s in the system.

Security Class Restrictions

LOGOUT may only be used to terminate logins belonging to the current *loginname*, unless the current *loginname* has sufficient security classification to terminate logins belonging to other *loginnames*. Administrator class and Operator class have full access to all LOGOUT functions. User class and Guest class may only use LOGOUT functions on logins that belong to the current *loginname*.

Examples

example 1: no options
LOGOUT terminates the current login.

```
LOGOUT
```

example 2: *username*
LOGOUT terminates all logins belonging to *loginnames* with the *username* BRAD, regardless of *groupname*.

```
LOGOUT \BRAD
```

example 3: *loginname*
LOGOUT terminates all logins with the *loginname* BRAD.WRKGRP.

```
LOGOUT \BRAD.WRKGRP
```

example 4: loginID
LOGOUT terminates *loginID* 12.

```
LOGOUT 12
```

example 5: fully specified loginID

LOGOUT terminates *loginID* 10 belonging to the *loginname* of BRAD.WRKGRP.

```
LOGOUT  \BRAD.WRKGRP\10
```

example 6: *terminalname*

LOGOUT terminates all logins at terminal TERM04.

```
LOGOUT  \TERM04
```

example 7: wildcards

LOGOUT terminates all logins under *loginnames* with the groupname WRKGRP (if the user has the security classification to do so).

```
LOGOUT  \*.WRKGRP
```

MKDIR (Make Directory)

Description

Creates a directory. You can abbreviate MKDIR as MD.

Syntax

MKDIR [*drive*:][*path*]*directoryname*
[[*drive*:][*path*]*directoryname* [...]]

Parameter

directoryname

Specifies the name of the new directory. Unless you specify a different *drive* and/or *path*, MS OS/2 *MULTIUSER* creates the new directory as a subdirectory of the current directory.

Remarks

You can create more than one directory at a time by specifying multiple directory names. MS OS/2 *MULTIUSER* creates each directory for which you do not specifically indicate a *drive* and *path* as a subdirectory of the current directory.

You must have Create (C) permission in the target directory.

Example

To create the directory CLIENT on drive A and the subdirectory DAVID under it, type the following:

```
MKDIR  A:\CLIENT  A:\CLIENT\DAVID
```

MODE (Set Device Mode)

Description

Sets operating parameters for communication and output devices that you may connect to or add to your computer. These devices include parallel and serial printers, modems, and screens. The MODE utility enables you to change settings by using a command line instead of physically setting switches in your computer.

The MODE utility has several different purposes. The following sections explain the different ways in which you can use the utility.

MODE: CONFIGURING A PARALLEL PRINTER

Description

Controls the line and character spacing of the output on a parallel printer.

Syntax

MODE LPT*n*[:] [*chars*][,[*lines*][,P]]

Parameters

n

Specifies the number of the parallel port the printer is connected to (1, 2, and so forth). (You can use PRN in place of LPT1; they are equivalent.) The colon is optional. This information is required.

chars

Specifies the number of characters per line. This number can be either 80 or 132; the default value is 80. If you enter nothing for this parameter, the MODE utility does not change the current number of characters per line.

lines

Specifies the number of lines per inch. This number can be either 6 or 8; the default value is 6. If you enter nothing for this parameter, the MODE utility does not change the current vertical spacing. You must type the comma before *lines*, even if you did not specify a new value for *chars*. If you do not specify a value for chars, do not include a space after LPT*n*[:].

P

Turns on "infinite retry," which causes MS OS/2 *MULTIUSER* to keep trying to send output to the printer if a time-out error occurs. With this option, part of the MODE utility remains resident in memory. You must type both commas before P, even if you did not specify new values for *chars* and *lines*. If you do not specify this option, infinite retry is turned off. If you do not specify a value for *chars*, do not include a space after LPT*n*[:].

Remarks

If your system gets stuck in a time-out loop where it keeps trying to send output to the printer but cannot succeed, you can exit from the loop by pressing **CTRL+C**.

The P parameter of the MODE utility will affect your printer only if the spooler is not running.

When you change the parallel printer mode, you are changing the default mode of the system. Users will print in this mode unless they override the default for their loginname using the CHANGE PRNMODE utility.

Security Class Restrictions

Only Administrators and Operators can change the parallel printer mode.

Example

To configure the printer connected to your computer's second parallel-printer port (LPT2) to print at 80 characters per line and 8 lines per inch, type the following:

```
MODE LPT2 80,8
```

MODE: CONFIGURING A SERIAL PORT

Description

Sets the parameters for communication with a serial printer or other device that uses a serial port. This is the port you use for asynchronous communication. Select a communication port that does not have a terminal configured for it.

Syntax

MODE COM*m*[:] [*baud*[,*parity*][,*databits*][,*stopbits*]]
[,TO=*state*][,XON=*state*][,IDSR=*state*][,ODSR=*state*]
[,OCTS=*state*][,DTR=*state*][,RTS=*state*][,BUFFER=*state*]

Parameters

m

Specifies the number of the serial port you want to use (1, 2, and so forth). The colon is optional. This information is required.

If you omit any of the following four parameters, the MODE utility uses the most recent settings. (The first time you use the utility, MODE uses the default settings; thereafter, it uses the most recent settings.) The MODE utility recognizes these arguments by their positions, so if you leave out an argument, you must still type the comma that precedes the next one.

baud

Specifies at least the first two digits of the transmission rate in bits per second: 110, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, or 19200. This argument is required, unless you are merely checking the current settings.

parity

Specifies how the system uses the parity bit to check for errors in transmission. The possible values are N (no parity), O (odd parity), E (even parity), M (mark; parity bit always 1), and S (space; parity bit always zero). The default value is E.

databits

Specifies the number of data bits in a character. This number can be 5, 6, 7, or 8; the default value is 7.

stopbits

Specifies the number of stop bits that define the end of a character. This number can be 1, 1.5, or 2. If the baud rate is 110, the default value for stopbits is 2; otherwise, the default value is 1. If you specify 1.5 for stopbits, you must specify 5 for databits.

Options

You can list the following options in any order after the *stopbits* argument, separating them with commas. The default settings listed for these options apply only when you first start your computer; thereafter, the default setting for an option is its previous setting.

TO=ON | OFF

Specifies whether infinite time-out processing is enabled (ON) or disabled (OFF). The default setting is OFF.

XON=ON | OFF

Specifies whether the XON/XOFF protocol for data-flow control is enabled (ON) or disabled (OFF). The default setting is OFF.

IDSR=ON | OFF

Specifies whether input handshaking that uses the DSR (Data Set Ready) circuit is enabled (ON) or disabled (OFF). The default setting is ON.

ODSR=ON | OFF

Specifies whether output handshaking that uses the DSR (Data Set Ready) circuit is enabled (ON) or disabled (OFF). The default setting is ON.

OCTS=ON | OFF

Specifies whether output handshaking that uses the CTS (Clear to Send) circuit is enabled (ON) or disabled (OFF). The default setting is ON.

DTR=ON | OFF | HS

Specifies whether the DTR (Data Terminal Ready) circuit is enabled (ON) or disabled (OFF), or whether DTR handshaking is enabled (HS). The default setting is ON.

RTS=ON | OFF | HS | TOG

Specifies whether the RTS (Request to Send) circuit is enabled (ON) or disabled (OFF), whether RTS handshaking is enabled (HS), or whether RTS toggling is enabled (TOG). The default setting is ON.

The following option applies only to communications hardware or drivers capable of extended hardware buffering.

BUFFER=ON | OFF | AUTO

Specifies whether extended hardware buffering is enabled (ON), disabled (OFF), or controlled automatically by the device driver (AUTO). The default setting is AUTO for device drivers that support extended hardware buffering; the default setting is OFF for device drivers that do not. (If a serial port does not support extended hardware buffering, the MODE utility does not display or change the setting for this option.) On a system where both the hardware and the device drivers do support extended hardware buffering, BUFFER=OFF can seriously slow down system performance; BUFFER=ON gives you the best possible performance on such a system, but this setting may not be compatible with some devices that have particularly strict requirements for the timing of events. Unless you are certain that your hardware supports extended hardware buffering and that you are using device drivers that also support it, do not change this option.

Remarks

If you type MODE COM*m* by itself, the utility displays the current settings for the specified serial port.

Some programs require additional information that you can specify with the MS OS/2 *MULTIUSER* version of the MODE utility. If you are going to use one of those programs, you must configure your serial port from the MS OS/2 *MULTIUSER* command prompt.

Example

To set the baud rate to 300, set the parity to odd, leave the number of data bits set to 7, and set the number of stop bits to 2 for your computer's first serial port, type the following:

```
MODE COM1: 300,O,,2
```

MODE: SETTING UP THE SCREEN

Description

Sets the way text is displayed on your screen, including the number of characters per line, the number of lines per screen, and whether or not the text is in color.

Syntax

MODE *display*[,*rows*]

Parameters

display

Specifies the kind of *display* adapter you are using: 40, 80, BW40, BW80, CO40, CO80, or MONO. For each of these options, 40 and 80 indicate the number of characters per line. BW means that color has been disabled even though you have a color graphics adapter; CO means that color has been enabled. MONO specifies a monochrome display adapter, which always has 80 characters per line.

rows

Specifies the number of *rows* (lines) on the screen: 25, 43, 50, or 60. Which of these is valid depends on the kind of display adapter you have. The default setting when you first start your computer is 25; thereafter, the previous setting is the default setting.

Remarks

When you use the MODE utility to change the way text is displayed on the screen, it affects only the current session. Refer to the Appendix D in the *Citrix MULTIUSER System Administrator's Guide* or Appendix A in the *Citrix MULTIUSER User's Guide*. See also the section in Chapter 3, "Setting the Display Mode," in the *Citrix MULTIUSER User's Guide*.

Example

To set a terminal so that it uses color, 80 characters per line, and 43 lines per screen (assuming the terminal can handle such settings), type the following:

```
MODE CO80,43
```

MODE: SETTING FLOPPY DISK VERIFY CAPABILITY

Description

Specifies whether MS OS/2 *MULTIUSER* should verify that data is correctly written to a floppy disk.

Syntax

```
MODE DSKT [VER=ON | OFF]
```

Remarks

The default setting is VER=OFF.

If you type MODE DSKT by itself, the utility displays the current setting.

This command affects all user logins.

MORE (Screen Pause)

Description

Reads from standard input and displays what it reads, one screenful at a time.

Syntax

MORE<*source*

Parameter

source
Specifies the *source* of the input. You can redirect input from a file or from an MS OS/2 *MULTIUSER* command or utility. For more information about redirection, see the *Citrix MULTIUSER User's Guide.*

Remarks

You can use MORE to view the contents of a long file or the results of a command screenful by screenful. After each screenful, MORE displays the following at the bottom of your screen:

```
--More--
```

You can press any key to display the next screenful.

Examples

To view the contents of the file IDEAS.OLD one screenful at a time, type the following:

```
MORE < IDEAS.OLD
```

To sort the file IDEAS.OLD alphabetically and view the output one screenful at a time, type the following:

```
SORT < IDEAS.OLD | MORE
```

MOVE (Move Files)

Description

Moves a file or directory from one directory to another on the same drive.

Syntax

MOVE [*drive*:][*path1*]*source* [*path2*][*destination*]

Parameters

source

Specifies the file or directory to be moved. You can use wildcard characters to specify a group of files, but you can move only one directory at a time. If you do not specify a *drive* or directory for *source*, MS OS/2 *MULTIUSER* searches for it in the current directory on the current drive.

destination

Specifies where the file or directory is to be moved. If you are moving a file, the *destination* can be either a file or a directory. The *destination* must always be on the same drive as the *source.*

Remarks

You can change the name of a file or directory as you move it, by specifying a new name in *destination.* If you are moving multiple files and you want to change their names, you must specify multiple destination file names by using the same wildcard characters you used in the *source file names.*

NOTE: In the High-Performance File System (HPFS), you can use the MOVE command to change the case of a *filename* or directory name by moving the file or directory to itself. For example, you can change the filename *mytaxfile.txt* to *MYTAXFILE.TXT* by using the uppercase version of the name as the *destination* argument. For more information about HPFS, see the *Citrix MULTIUSER User's Guide.*

When you use the MOVE command to relocate a file or directory, MS OS/2 *MULTIUSER* will not write over an existing file or directory that has the same name. If you specify an existing *filename* as the *destination*, MS OS/2 *MULTIUSER* displays an error message and does not carry out the command. If you specify an existing directory name as the *destination*, MS OS/2 *MULTIUSER* moves the *source* file or directory into that directory; if the *source* was a directory, it becomes a subdirectory of the *destination* directory.

Moving a file or directory does not change its date or time.

You must have Read (R) permission on the *source* file/directory and Create (C) permission on the *destination* file/directory.

Examples

To move all the files with the extension *.DOC* from the current directory to the directory \REPORTS\OLD on the current drive and change their extensions to *.OLD*, type the following:

```
MOVE *.DOC \REPORTS\OLD\*.OLD
```

To move the directory LETTERS from the current directory and make it a subdirectory of the directory \BUSINESS\SEPT89 on the current drive, type the following:

```
MOVE LETTERS \BUSINESS\SEPT89
```

MSG (Send a Message)

Description

Sends a message to a user.

Syntax

MSG [*username*[*.groupname*]] [/TIME:*seconds*] [/V] [/?] [*message*]

MSG [\ *username.groupname*\ *loginID*] [/TIME:*seconds*] [/V] [/?] [*message*]

MSG [\ *terminalname*] [/TIME:*seconds*] [/V] [/?] [*message*]

Parameters

username[*.groupname*]
: Identifies all logins belonging to the specified *username* and the specified *groupname*. If the *groupname* is not specified, all *groupnames* for that *username* are used (equivalent to *username.∗*).

\ *username.groupname*\ *loginID*
: Identifies a specific login belonging to the specific *loginname.*

\ *terminalname*
: Identifies all logins connected to *terminalname.*

message (message string)
: The text of the message to send. If *message* is not specified, standard input (STDIN) is read for the message.

Options

/TIME:*seconds* (time delay)
The amount of time to wait for an acknowledgement from the target login that the message has been received.

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

If no *username* or *terminalname* is specified, MSG displays an error message. If the *terminalname* is specified, the *terminalname* must identify an active terminal. Wildcards may be used in the input parameter specifying the *username* or *loginID.* If a single wildcard is present (*), it is interpreted as a wildcard for the *username* and specifies all *usernames* in the system.

Examples

example 1: *username*
MSG sends the message "this is a message" to all logins belonging to all *loginnames* with *username* BRAD. The following two examples are equivalent.

```
MSG  BRAD  this is a message

MSG  BRAD.*  this is a message
```

example 2: *loginname*

MSG sends the message "let's meet as soon as possible today" to all logins belonging to the *loginname* BRAD.WRKGRP. The following two examples are equivalent.

```
MSG  \BRAD.WRKGRP  let's meet as soon as possible today
```

```
MSG  BRAD.WRKGRP  let's meet as soon as possible today
```

example 3: loginID

MSG sends the message "call John at 11:00" to *loginID* 12 belonging to the *loginname* BRAD.WRKGRP. The following two examples are equivalent.

```
MSG  \BRAD.WRKGRP\12  call John at 11:00
```

```
MSG  .\12  call John at 11:00
```

example 4: *terminalname*

MSG sends the message "logout as your terminal will be reset" to all logins at terminal TERM05.

```
MSG  \TERM05  logout as your terminal will be reset
```

example 5: message text in a file

MSG sends the text in the file *MSG.DAT* as a message to all logins belonging to the *loginname MIKE.WRKGRP.*

```
MSG  \MIKE.WRKGRP  <  MSG.DAT
```

example 6: wildcards

MSG sends the message "the meeting will start in 5 minutes" to all users logged into the system.

```
MSG * the meeting will start in 5 minutes
```

example 7: timeout option

MSG sends the message "coffee break!?!" to all *loginnames* in group WRKGRP, with an acknowledgement timeout of 10 seconds. The following two examples are equivalent.

```
MSG \*.WRKGRP /T:10 coffee break!?!
```

```
MSG .WRKGRP /T:10 coffee break!?!
```

OWNER (Ownership Control)

Description

This utility is used to display and change the owner designation on HPFS files and directories.

Syntax

OWNER *pathname* [*username*[.*groupname*]] [/C:*username*[.*groupname*]] [/S] [/DIR] [/FILE] [/?]

Parameters

pathname

This is a filename or directory name that you wish to query or change. You can enter wildcards to operate on multiple files.

username[.*groupname*]

This is used to select and display only files and directories owned by this *loginname*. The *loginname* can be specified as user.* or *.group. If the *groupname* is not specified, all *groupname*s for that *username* are used (equivalent to *username.**). If this parameter is left blank, a *loginname* of *.* is assumed. You can also enter UNOWNED to select unowned files and directories.

Options

/C:*username*[.*groupname*]

Specifies an owner change operation. The files and directories selected by *pathname* and *loginname* are changed to the new loginname specified by this option. If *groupname* is not given, default group is used. No wildcards are accepted.

/S

Specifies that owner should operate on the entire subdirectory tree below that which was given in *pathname*.

/DIR

Specifies that OWNER only should operate on directories.

/FILE

Specifies that OWNER only should operate on files.

/? (help)

Displays the syntax for the utility and information about the utility's options.

Remarks

The owner value is important in the ability to modify the security attributes of files and directories.

OWNER works only for HPFS partitions.

It is possible to have unowned files and directories; this occurs after deleting a user who had owned files. An Administrator can find these by specifying UNOWNED as the *loginname.*

If neither /FILE or /DIR is specified, both are implied.

Security Class Restrictions

Any class can query ownership of files and directories they can read. In addition, the owner can query the list of files and directories that they own regardless of class.

Operators and Administrators can query other owners file lists.

The owner can change (relinquish) ownership of a file or directory.

Administrator class can change any owner value.

Examples

OWNER MYDIR
    Displays the owner loginname for the directory MYDIR.

OWNER MYDIR*.* DAVIDH
    Displays all the files and directories owned by username DAVIDH in all groups.

OWNER MYDIR*.* *.CITRIX /S
Displays all the files and directories at and below MYDIR owned by all users in group CITRIX.

OWNER MYDIR /S
Displays all the files and directories at and below MYDIR, including MYDIR, owned by all users.

OWNER MYDIR UNOWNED /C:ANDYS.CITRIX /S
Changes all the files and directories at and below MYDIR, including MYDIR, that are unowned to be owned by *loginname* ANDYS.CITRIX.

OWNER MYDIR\FILE*.TXT /C:ANDYS
Changes all the files in directory MYDIR which match the description "FILE*.TXT" to be owned by ANDYS.WRKGRP.

NOTE: WRKGRP is the default login group for ANDYS.

OWNER MYDIR*.* /FILE

Displays the owner *loginname* for all the files in the directory MYDIR.

PASSWORD (Change Password)

Description

Changes the PASSWORD for the default or specified *loginname.*

Syntax

PASSWORD [*username*[.*groupname*]]
[/PASSWORD:*old*/*new*/*new*] [/Q] [/?]

Parameter

username[.*groupname*]

Specifies the *loginname* for which the password is to be changed. If this is not given, the current *loginname* is used. If only *username* is given, the *groupname* is the default group for that user. Wildcards are not supported. See Security Class Restrictions below.

Options

/PASSWORD

Used to specify the *old* and *new* passwords on the command line. Omitting this option causes PASSWORD to prompt for the *old* and *new* passwords.

/Q

Used to query the expiration date of the password for the default or specified *loginname*.

/? (help)

Displays the syntax for the utility and information about the utility's options.

Remarks

If the PASSWORD command is entered without a *loginname*, PASSWORD changes the password for the current *loginname*. If you enter PASSWORD without the /PASSWORD or /Q options, you are prompted for the old password, the new password, and verification for the new password. If you do not enter a new password and press **ENTER** instead, your password will be deleted.

By default, the password must begin with A-Z or a-z; the rest of the password can include numbers (0-9). It must be at least six characters long and cannot be equal to one of the last four passwords you have used. The password is case sensitive.

The System Administrator can change the password requirements and can place an expiration time on the passwords. If you have a problem at login, contact your System Administrator.

Security Class Restrictions

Any class may change his or her own password (the password for the current loginname). Only Administrator class can change a password for another loginname.

Examples

Example 1: no options
Change the password for the current *loginname*.

```
PASSWORD
Old Password: BRUTIS
New Password: SLP
New Password: SLP
```

Example 2: *loginname* only
Change the password for the *loginname* specified.

```
PASSWORD  DAVIDH.WRKGRP
Old Password: BRUTIS
New Password: SLP
New Password: SLP
```

Example 3: *loginname* and password
Change the password to the specified password for the specified *loginname.*

```
PASSWORD  DAVIDH.WRKGRP
/PASSWORD:BRUTIS/SLP/SLP
```

Example 4: query option only
Query the expiration date for the current *loginname.*

```
PASSWORD  /Q
```

Example 5: *loginname* and query option
Query the expiration date for the specified *loginname.*

```
PASSWORD  SHERYLP.WRKGRP  /Q
```

PATCH (Apply System Fix)

Description

Inserts a section of program code into an existing MS OS/2 *MULTIUSER* program or application to change the way the program runs. Any file that can be written to can be patched.

Syntax

PATCH [*drive*:][*path*]*filename* [/A]

Parameter

filename
Specifies the file to PATCH.

Option

/A

Specifies the automatic operation mode. With the /A option, *filename* must be a file containing instructions for patching one or more files automatically.

Remarks

The PATCH utility has two modes of operation: automatic and interactive. Interactive mode is the default mode. In this mode, you supply the path of the file you want to patch on the PATCH command line. PATCH then prompts you for the offset at which a patch is to be made and for the patch contents. (PATCH can change bytes at any position in a file or add bytes to the end of a file.) You must type both the offset and the patch contents in hexadecimal notation.

After you supply the hexadecimal offset, the PATCH utility displays the 16 bytes at that offset. You can then change any or all of the 16 bytes. If you decide not to make any changes, you can press the **ESC** key.

The cursor is initially positioned on the first byte. To change this byte, type one or two hexadecimal digits. To leave the byte unchanged and move to the next byte, press the **SPACEBAR**. Press **BACKSPACE** to move the cursor back if you make a mistake. If you move the cursor past the last byte displayed, the PATCH utility displays the next 16 bytes. This cycle continues until you press **ENTER**.

When you press **ENTER**, the PATCH utility saves the patch information and asks if you want to make any more patches. If you respond with "Y", PATCH again prompts you for an offset. After you have entered all the patches you want to make and responded with "N" at the "more patches" prompt, PATCH displays the patches on the screen and asks if they should be applied. If you respond with "Y", PATCH writes all the saved patch requests to disk in the same order in which you entered them.

You must have Write (W) permission on the file to be patched.

WARNING: You should use the PATCH utility only if you understand the need for a patch, how to make the patch, and the effect the patch will have on program operation. Before you use the PATCH utility, be sure to make backup copies of the files to which the patches will be applied.

PATH (Set Search Path for Programs)

Description

Specifies where MS OS/2 *MULTIUSER* is to search for a command file or application if the program is not in the current directory.

Syntax

PATH [; | [*drive*:]*path*[;...]]

Parameters

;

When used alone, (PATH ;) clears all search-path settings. Semicolons are also used to separate multiple paths.

path

Specifies the *path* of the directory that MS OS/2 *MULTIUSER* is to search for command files. If you do not specify a *drive*, MS OS/2 *MULTIUSER* uses the current drive.

Remarks

If any name in one of the paths you set contains a semicolon (;), you must enclose each *path* you specify in double quotation marks (").

If you type PATH by itself, MS OS/2 *MULTIUSER* displays the current search path.

The PATH command affects only the current session. To specify a *path* for all MS OS/2 *MULTIUSER* sessions, you must set the PATH environment variable in your *CONFIG.SYS* file. For information about how to do this, see the SET command.

If you do not have Execute (X) access to a path that is in your PATH environment variable, that path will be skipped during search operations.

Example

To instruct MS OS/2 *MULTIUSER* to search for a command file in the directories BIN and WORD on drive C and the subdirectory WORD of the APPS directory on drive D, type the following:

```
PATH C:\BIN;C:\WORD;D:\APPS\WORD
```

To instruct MS OS/2 *MULTIUSER* to append a directory C:\USR\CHRISL to the current path, enter the following:

```
PATH %PATH%C:\USR\CHRISL
```

The %PATH% tells MS OS/2 *MULTIUSER* to use the path that is currently defined.

PAUSE (Pause Batch Processing)

Description

Suspends processing of a batch file and displays a message that asks you to press any key to continue. Use this command only in a batch file.

Syntax

PAUSE [*comment*]

Parameter

comment
Specifies the message you want to have appear before the line "Press any key when ready...". The message can be any combination of characters. The echo feature must be on for *comment* to appear on the screen.

Remarks

Use this command to make a batch program pause any time you have to do something, such as read a message or insert a disk, before continuing with the program.

Example

To have your batch program pause and prompt you to insert a new disk in drive A before continuing, type the following at the appropriate point in the batch file:

```
PAUSE  Please put a new disk into drive A.
```

PRINT (Print Files)

Description

Sends a file to be printed to a device.

Syntax

PRINT [/D:*device*] [/B] [*drive*:][*path*]*filename* [...]

or

PRINT [/D:*device*] [/T | /C]

Parameter

filename
: Specifies the file to print. You can specify more than one file to print.

Options

/D:*device*
: Specifies the printer to use. The default device is LPT1.

/B
: Instructs MS OS/2 *MULTIUSER* not to interpret **CTRL+Z** characters in the file as end-of-file characters.

/T

Stops all printing and removes all files from the print queue if the spooler is active for the specified device.

/C

Stops printing the file being printed and removes it from the print queue if the spooler is active for the specified device.

Security Class Restrictions

Administrators and Operators can remove or stop any print files. Users and Guests can remove or stop only their files from the print queue.

Example

To print the file *PENCIL.TST* (which is in the current directory on the current drive) on the printer connected to your computer's third parallel port (LPT3), type the following:

```
PRINT /D:LPT3 PENCIL.TST
```

PROMPT (Change Command Prompt)

Description

Changes the command prompt for the current MS OS/2 *MULTIUSER* command interpreter. This change affects only the current session.

Syntax

PROMPT [*string*]

Parameters

string

Specifies the new prompt. You can specify any character *string* you want or you can use any of the $x character combinations from the list that follows to customize your prompt. (You can combine text and these character combinations in any order.) Any spaces that you type appear as part of the prompt.

Remarks

The following character combinations are available for use in customizing your prompt:

Combination	Action
$$	Displays the dollar-sign character ($).
$_	Begins a new line on the screen.
$a	Displays an ampersand (&).
$b	Displays the pipe symbol (|).
$c	Displays a left parenthesis [( ].
$d	Displays the current date.
$e	Specifies the beginning of an ANSI escape code.
$f	Displays a right parenthesis [ )].
$g	Displays the greater-than sign (>).

Combination	Action
$h	Represents the backspace character; this erases the previous character from the prompt.
$i	Displays the Help line.
$l	Displays the less-than sign (<).
$n	Displays the current drive letter.
$o	Displays the current hostname.
$p	Displays the current drive letter and path.
$q	Displays the equal sign (=).
$t	Displays the current time.
$u	Displays the current username.
$v	Displays the MS OS/2 *MULTIUSER* version number.

If you type PROMPT by itself, MS OS/2 *MULTIUSER* resets the prompt to the system default prompt. The system default prompt is the current *drive* letter and the name of the current directory, in brackets.

The PROMPT command affects only the current session. To specify a prompt for all MS OS/2 *MULTIUSER* sessions, you must set the PROMPT environment variable in your *CONFIG.SYS* file. For more information about how to do this, see the SET command.

Example

To change your prompt so that it displays the time on one line and the current drive and directory on the next, followed by a space and a greater-than sign, type the following:

```
PROMPT  The time is $t$h$h$h$h$h$h$_$p $g
```

The new prompt has the following form:

```
The time is 13:37
C:\OS2 >
```

Notice that the $h characters erases the seconds and hundredths of seconds from the time display.

PSTAT (Process Status Information)

Description

Displays information about the status of one or more processes in the system.

Syntax

PSTAT [/C | /S | /L | /M | /P:*processID*]

Options

/C

Displays information about each of the current processes and threads.

/S

Displays system-semaphore information for each thread in the system.

/L

Displays the names of the run-time linked libraries for each process in the system.

/M

Displays the named-shared memory information for each process in the system.

/P:*processID*

Displays information about a particular process, identified by its process identification number (*processID*). The number must be hexadecimal so /P:10 refers to process 16 (decimal).

Remarks

If you type PSTAT by itself, the utility displays all the information for all the options.

Security Class Restrictions

Only Administrator and Operator classes are allowed to use this utility.

QUERY (View Current Status)

Description

The QUERY utilities are used to display current information about the system such as the current allocation of resources and system status. The information that the QUERY utilities display is a snapshot of the system at the time when the utility executes.

Syntax

The command "QUERY" can invoke any one of the query utilities. The command line format is:

```
QUERY [AUDIT | EVENTS | HOST | LIMITS | LOGIN |
MEMORY | PRINT | PROCESS | SESSION | TERMINAL | USER]
[/?]
```

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

QUERY AUDIT

Description

Displays the system-wide security audit events that are currently being logged.

Syntax

```
QUERY AUDIT [/?]
```

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

This only queries the system-wide auditing flags. Audit flags can also be placed on users and resources using CONFIG USER and CONFIG ACCESS. These commands must be used to query the user and resource audit flags.

Auditing can be turned on and off by using CHANGE EVENTS. If auditing is off, the audit flags displayed here will not be used until audit event logging is turned on.

Security Class Restrictions

This utility is available only to an Operator or Administrator.

QUERY EVENTS

Description

Displays which events are currently being logged.

Syntax

```
QUERY EVENTS [/?]
```

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

QUERY EVENTS displays the current state of event logging. The events being logged can be changed permanently using CONFIG SYSTEM. Event logging can also be controlled using CHANGE EVENTS.

Security Class Restrictions

This utility is available only to an Operator or Administrator.

QUERY HOST

Description

Displays information about the current MS OS/2 *MULTIUSER* system.

Syntax

```
QUERY HOST [/LICENSE] [/V] [/?]
```

Options

/LICENSE (licensed number of logins)
Displays information about the licensed number of logins permitted on the system.

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

QUERY HOST displays information about the host that the current user is logged into.

QUERY HOST returns the following information about the current system:

- Its hostname
- The total number of active terminals
- The total number of active logins
- The total number of sessions
- The total number of processes
- The startup time of the system.

The following is sample output for the QUERY HOST command:

```
HOSTNAME TERMINALS LOGINS SESSIONS PROCESSES    BOOT TIME
System1       9        9        21         28   06/20/91 06:20
```

HOSTNAME identifies the system. TERMINALS specifies the number of active terminals on the system. LOGINS specifies the total number of logins in the system. SESSIONS specifies the total number of sessions in the system. PROCESSES specifies the total number of processes in the system. BOOT TIME indicates the date and time the system was started.

Example

example 1

QUERY HOST displays information about the current system.

```
QUERY HOST
```

QUERY LIMITS

Description

QUERY LIMITS displays the user resource limits and current totals for one or more users in the system. The minimum and maximum limit and current total for each resource will be displayed. If no *loginname* is specified on the command line, the current *loginname* is used.

Syntax

QUERY LIMITS [*username*[*.groupname*]] [/?]

Parameter

username[*.groupname*]
Identifies the *loginname* whose resource limits are to be displayed. If the *groupname* is not specified, all *groupnames* for that *username* are used (equivalent to *username.**).

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

The default for the QUERY LIMITS command is to display the resource limits and current totals for the current *loginname*. If RESOURCE=OFF in *CONFIG.SYS*, the message "Resource Management is turned off" is displayed.

Security Class Restrictions

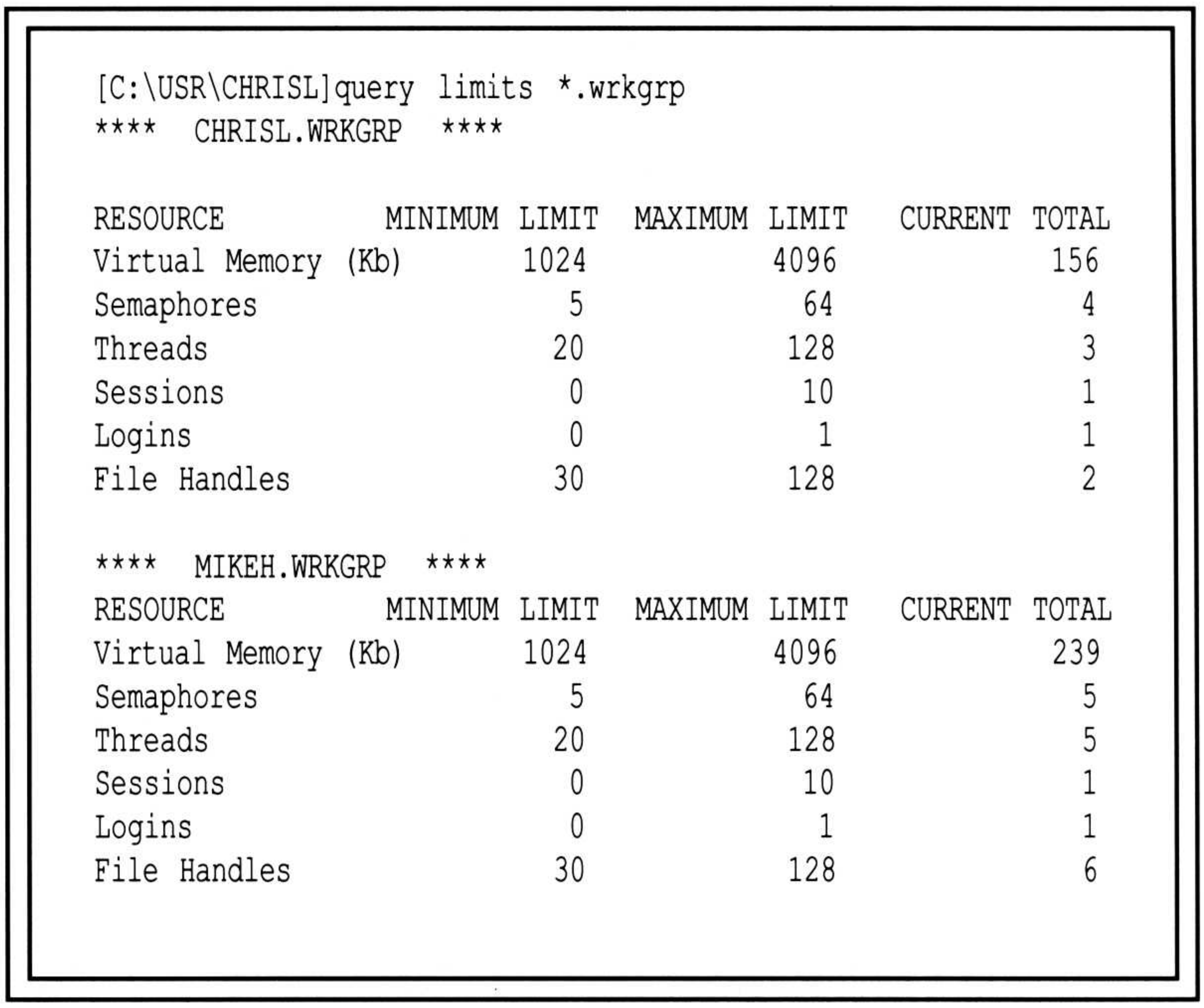
Administrators and Operators can query all user resource limits. Guests and Users can query their own resource limits.

Examples

The *loginname* can contain wildcard characters as in the following examples, provided the user has the proper security classification:

```
QUERY LIMITS  *.*
QUERY LIMITS  *.WRKGRP
QUERY LIMITS  MIKEH.WRKGRP
```

Sample output for QUERY LIMITS follows:



```
[C:\USR\CHRISL]query limits *.wrkgrp
****  CHRISL.WRKGRP  ****

RESOURCE          MINIMUM LIMIT  MAXIMUM LIMIT   CURRENT TOTAL
Virtual Memory (Kb)       1024            4096             156
Semaphores                   5              64               4
Threads                     20             128               3
Sessions                     0              10               1
Logins                       0               1               1
File Handles                30             128               2

****  MIKEH.WRKGRP  ****
RESOURCE          MINIMUM LIMIT  MAXIMUM LIMIT   CURRENT TOTAL
Virtual Memory (Kb)       1024            4096             239
Semaphores                   5              64               5
Threads                     20             128               5
Sessions                     0              10               1
Logins                       0               1               1
File Handles                30             128               6
```

QUERY LOGIN

Description

Displays information about logins.

Syntax

QUERY LOGIN [*loginID*] [/V] [/?]

QUERY LOGIN [\ *username.groupname*\ *loginID*] [/V] [/?]

QUERY LOGIN [\ *terminalname*] [/V] [/?]

Parameters

loginID
Identifies a login.

\ *username.groupname*\ *loginID*
Identifies a specific login belonging to the specified *loginname.*

\ *terminalname*
Identifies all logins connected to *terminalname.*

Options

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

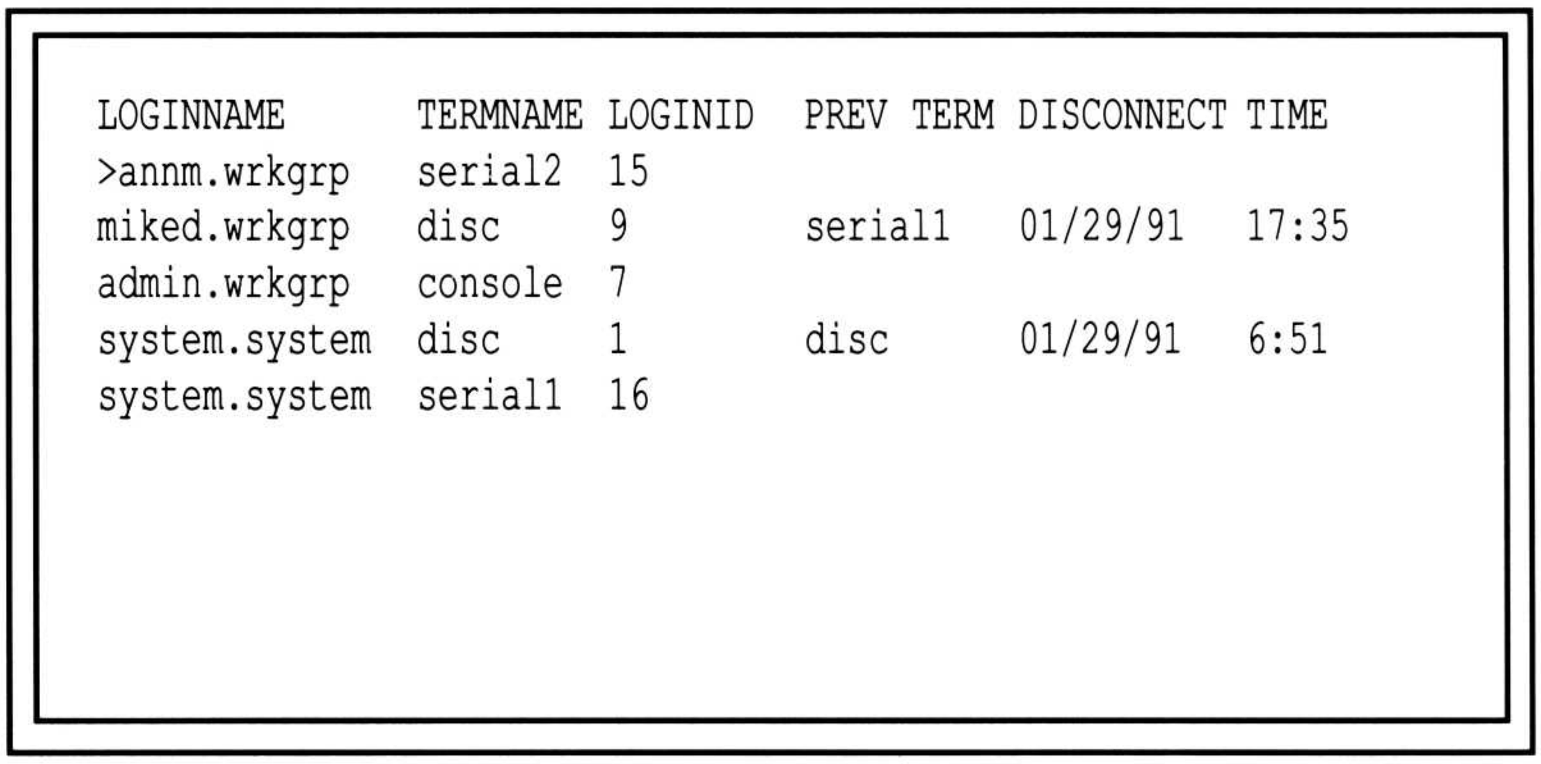
If no *loginID* or *terminalname* is specified, QUERY LOGIN displays all logins for the current *loginname.* If the *terminalname* is specified, the *terminalname* must identify an active terminal. Wildcards may be used in the input parameter specifying the *loginID.* If a single wildcard is present (*), it is interpreted as a wildcard for the *loginID* and specifies all *loginIDs* in the system.

QUERY LOGIN returns the following information:

- The *loginname* that owns the login
- The *terminalname* where the login is connected
- The *loginID* of the login

- The previous terminal the login was connected to, if different from the current terminal.
- The disconnect time, if the login is disconnected

Sample output for QUERY LOGIN follows:



```
LOGINNAME      TERMNAME LOGINID  PREV TERM DISCONNECT TIME
>annm.wrkgrp   serial2  15
miked.wrkgrp   disc     9        serial1   01/29/91   17:35
admin.wrkgrp   console  7
system.system  disc     1        disc      01/29/91   6:51
system.system  serial1  16
```

NOTE: The > character indicates the current login. LOGINNAME identifies the *username* and *groupname* of each login listed. TERMNAME identifies the *terminalname* where the login is connected. LOGINID identifies the login. PREV TERM specifies the *terminalname* at which the login was previously connected. DISCONNECT TIME specifies the time at which the login was disconnected.

Security Class Restrictions

QUERY LOGIN may only be used to query logins belonging to the current *loginname*, unless the current *loginname* has sufficient security classification to query logins belonging to other *loginnames.* Administrator class, Operator class, and User class have full access to all QUERY LOGIN functions. Guest class may only use QUERY LOGIN functions on logins that belong to the current *loginname.*

Examples

example 1: no options

QUERY LOGIN displays information about all logins belonging to the current *loginname.*

```
QUERY LOGIN
```

example 2: *username*

QUERY LOGIN displays information about all logins belonging to all *loginnames* with the *username* BRAD.

```
QUERY LOGIN  \BRAD
```

example 3: *loginname*

QUERY LOGIN displays information about all logins belonging to the *loginname* BRAD.WRKGRP.

```
QUERY LOGIN  \BRAD.WRKGRP
```

example 4: *loginID*

QUERY LOGIN displays information about loginID 12.

```
QUERY LOGIN  12
```

example 5: fully specified *loginID*

QUERY LOGIN displays information about loginID 10 belonging to the *loginname* of BRAD.WRKGRP.

```
QUERY LOGIN  \BRAD.WRKGRP\10
```

example 6: *terminalname*

QUERY LOGIN displays information about all logins at terminal SERIAL0.

```
QUERY LOGIN \SERIAL0
```

example 7: wildcards

QUERY LOGIN displays information about all logins in the system.

```
QUERY LOGIN *
```

QUERY MEMORY

Description

QUERY MEMORY displays information about the allocation of user application memory in the system. This information includes both the amount of memory present in the system and the amount of memory that is currently swapped to disk. There are additional options to get raw data, free physical memory, and dynamic link library memory usage.

Syntax

QUERY MEMORY [*username*[.*groupname*]]

QUERY MEMORY [*username*[.*groupname*]\ *loginID*\ *sessionID*\ *processID*] [/DLL] [/RAW] [/?]

QUERY MEMORY [\ *terminalname*] [DLL] [RAW] [/?]

QUERY MEMORY [/FREE] [/?]

Parameters

username[.*groupname*]

Identifies all processes of the specified *loginname*. If the *groupname* is not specified, all *groupname*s for that *username* are used (equivalent to *username.**).

username[.*groupname*]\ *loginID*\ *sessionID*\ *processID*

Identifies a specific process belonging to the specific user context (loginname, login, session). If the *groupname* is not specified, all *groupname*s for that *username* are used (equivalent to *username.**).

\ *terminalname*
Identifies all processes running at *terminalname* regardless of user context.

Options

/DLL
Memory usage for dynamic link libraries (for the user) is displayed.

/RAW
Provides low level details on each segment allocated for each process.

/FREE
Provides information on the amount of free physical memory in the system.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

The default for QUERY MEMORY is to display the process related memory allocations for the current *loginname*. If the DLL option is specified, memory usage for dynamic link libraries (in use for the user) is also displayed.

The information is broken down into two totals if the DLL option is NOT specified. The first set of totals relates to the private memory allocations. Private memory is the principal memory counted by resource management for your limits. The second set of totals relate to the shared memory allocated by the applications. This shared memory can be used by several users at once and are therefore not included in the resource management totals for an individual user.

If the DLL option is specified, a third total is also displayed. This sums up all of the dynamic link library allocations. Once again, since this memory is shared, it is not included in the resource management totals for an individual user.

If the FREE option is specified, a summary of free physical memory is displayed. This summary is broken down by size of Free Blocks and can be useful as an indication of fragmentation in physical memory. Fragmentation occurs when free memory is broken up into many small (<64 Kbytes) blocks.

While the QUERY MEMORY command gives you an idea of the amount of memory allocated on behalf of each user, the QUERY LIMITS command should be used to obtain the current usage of memory for a user.

Security Class Restrictions

Administrators and Operators can query memory for all users. Guests and Users can query their own memory usage.

Examples

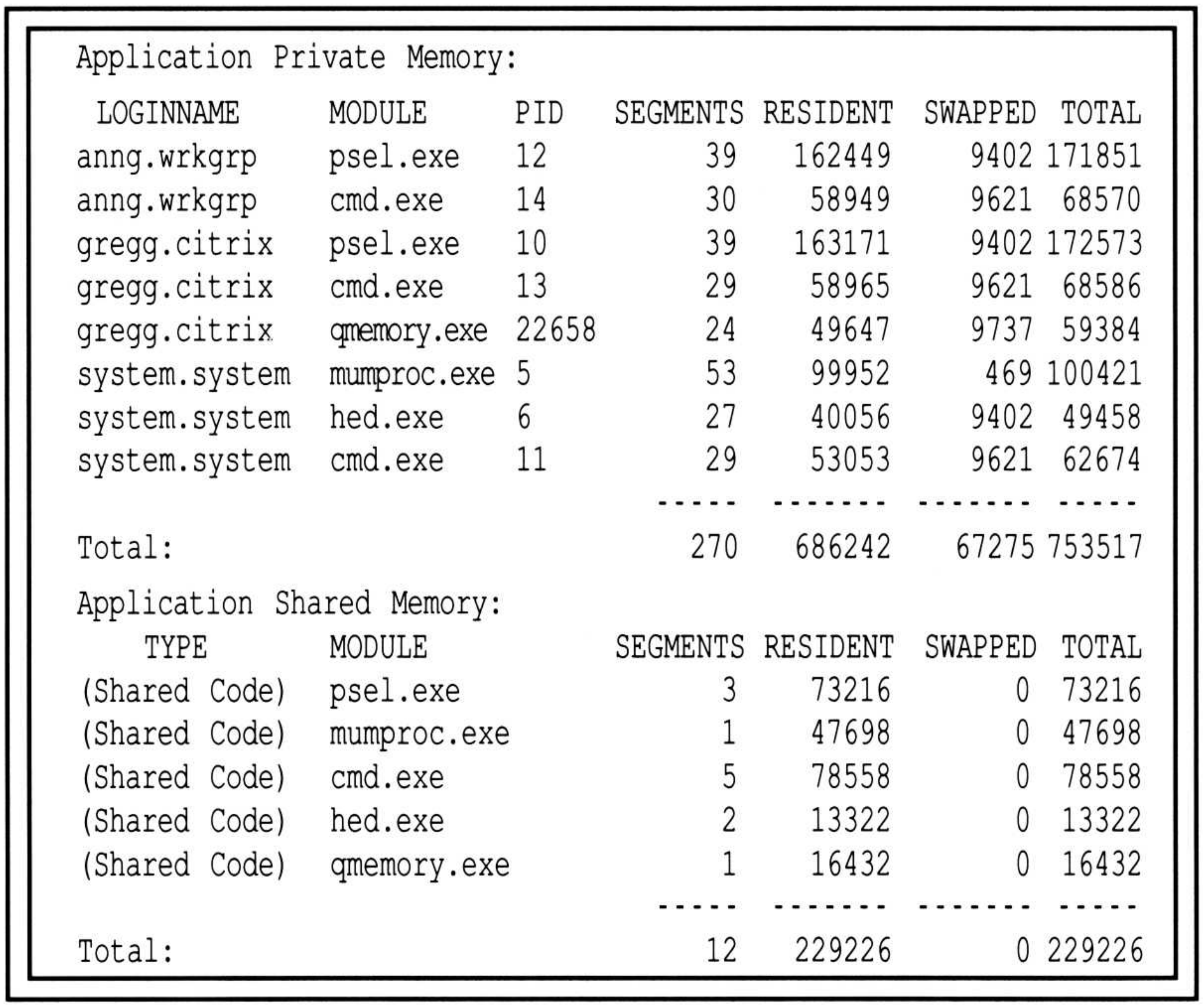
The *loginname* can contain wildcard characters as in the following examples, provided the user has the proper security classification:

```
QUERY MEMORY  *.* /DLL
QUERY MEMORY  *.WRKGRP
QUERY MEMORY  GREGG.*
QUERY MEMORY  GREGG.WRKGRP
```

Below is a description of the output from the command assuming an Administrator invoked the command:

QUERY MEMORY *.*

Sample output for QUERY MEMORY follows:



```
Application Private Memory:
 LOGINNAME     MODULE      PID    SEGMENTS RESIDENT  SWAPPED  TOTAL
anng.wrkgrp    psel.exe    12           39   162449     9402 171851
anng.wrkgrp    cmd.exe     14           30    58949     9621  68570
gregg.citrix   psel.exe    10           39   163171     9402 172573
gregg.citrix   cmd.exe     13           29    58965     9621  68586
gregg.citrix   qmemory.exe 22658        24    49647     9737  59384
system.system  mumproc.exe 5            53    99952      469 100421
system.system  hed.exe     6            27    40056     9402  49458
system.system  cmd.exe     11           29    53053     9621  62674
                                     -----  -------  ------- -----
Total:                                 270   686242    67275 753517
Application Shared Memory:
    TYPE       MODULE             SEGMENTS RESIDENT  SWAPPED  TOTAL
(Shared Code)  psel.exe                  3    73216        0  73216
(Shared Code)  mumproc.exe               1    47698        0  47698
(Shared Code)  cmd.exe                   5    78558        0  78558
(Shared Code)  hed.exe                   2    13322        0  13322
(Shared Code)  qmemory.exe               1    16432        0  16432
                                     -----  -------  ------- -----
Total:                                  12   229226        0 229226
```

QUERY PRINT

Description

QUERY PRINT displays the current print queues in the system and the print jobs in each queue for a specific *loginname*, if there are any.

Syntax

QUERY PRINT [*username*[*.groupname*]] [/?]

Parameter

username[*.groupname*]

The *loginname* for the user whose print jobs are to be displayed. If no *loginname* is specified, all print jobs are displayed. If the *groupname* is not specified, all *groupnames* for that *username* are used (equivalent to *username.**).

Option

/? (help)

Displays the syntax for the utility and information about the utility's options.

Remarks

The default for QUERY PRINT is to display the print jobs for the entire system.

The *loginname* can contain wildcard characters.

Security Class Restrictions

Administrators, Operators, and Users can query all print jobs in the system. Guests can only query their own print jobs.

Examples

To query all print jobs in the system:

```
QUERY PRINT
```

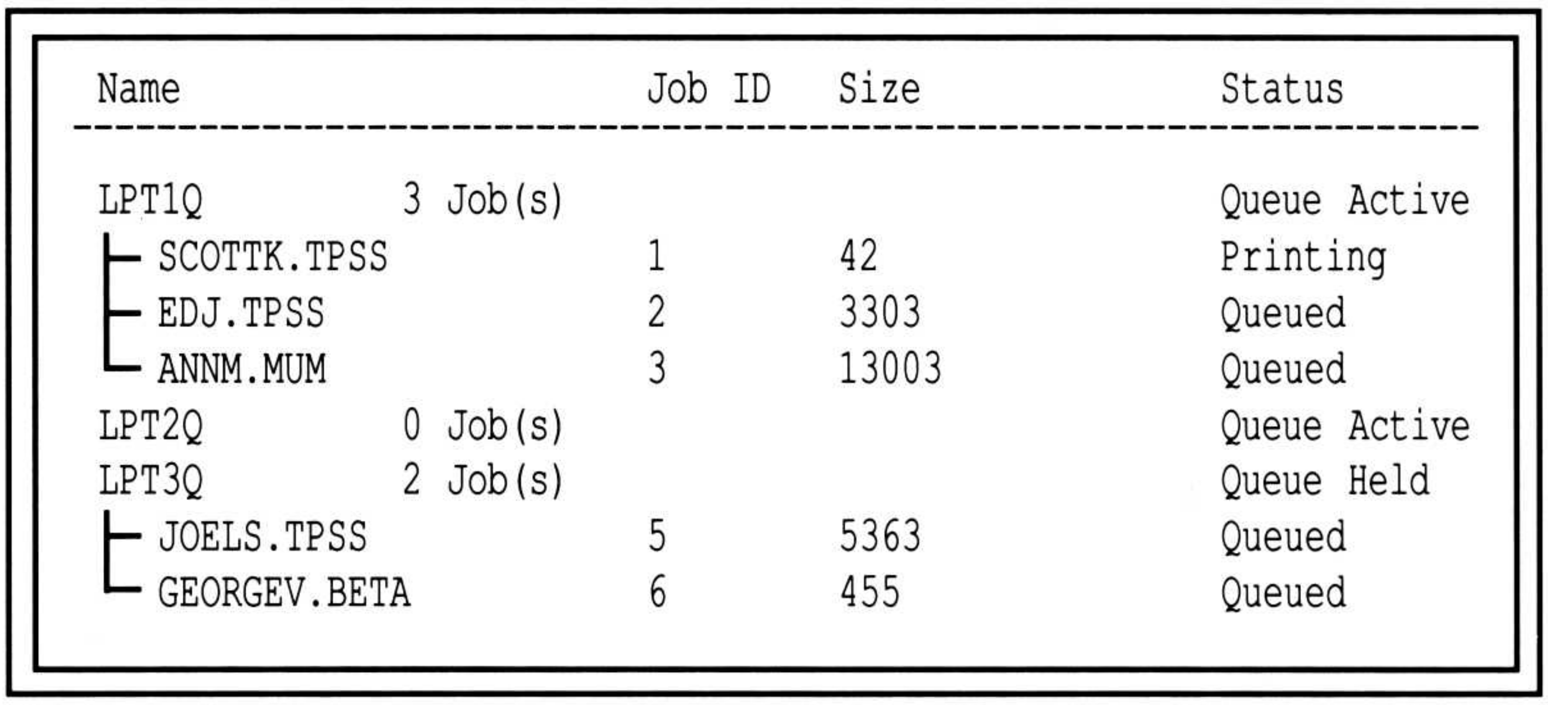
To query all print jobs belonging to users in group WRKGRP:

```
QUERY PRINT  *.WRKGRP
```

To query all print jobs for JEFFK in the group WRKGRP:

```
QUERY PRINT  JEFFK.WRKGRP
```

Sample output for QUERY PRINT follows:



```
Name                        Job ID   Size              Status
--------------------------------------------------------------------
LPT1Q           3 Job(s)                               Queue Active
├─ SCOTTK.TPSS              1        42                Printing
├─ EDJ.TPSS                 2        3303              Queued
└─ ANNM.MUM                 3        13003             Queued
LPT2Q           0 Job(s)                               Queue Active
LPT3Q           2 Job(s)                               Queue Held
├─ JOELS.TPSS               5        5363              Queued
└─ GEORGEV.BETA             6        455               Queued
```

QUERY PROCESS

Description

Displays information about processes.

Syntax

QUERY PROCESS [*processID*] [/V] [/?]

QUERY PROCESS
[\ *username.groupname*\ *loginID*\ *sessionID*\ *processID*]
[/V] [/?]

QUERY PROCESS [\ *terminalname*] [/V] [/?]

Parameters

processID
Identifies a process.

\ *username.groupname*\ *loginID*\ *sessionID*\ *processID*
Identifies a specific process belonging to the specific user context (loginname, login, session).

\ *terminalname*
Identifies all processes running at *terminalname* regardless of user context.

Options

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

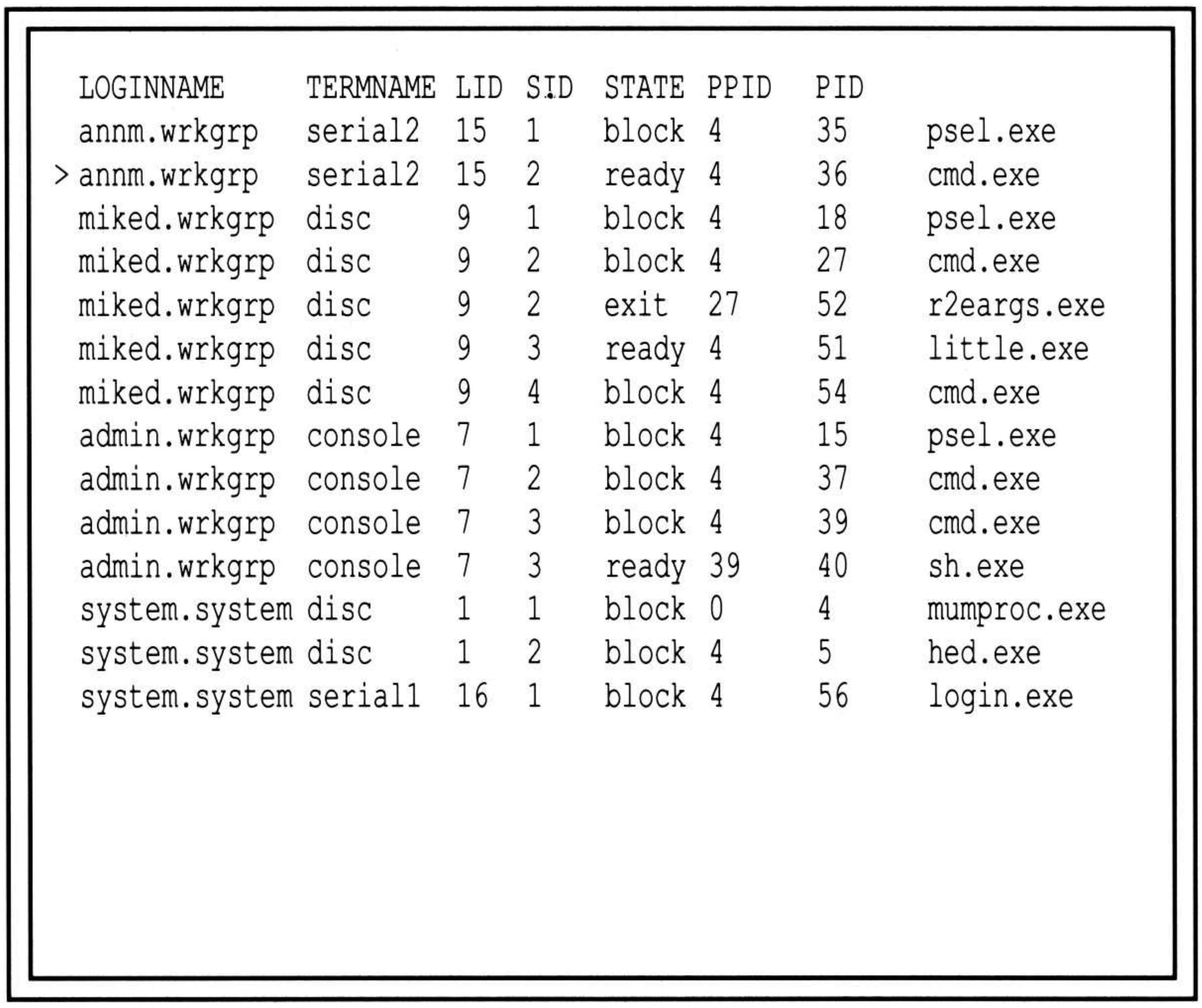
Remarks

If no *processID* or *terminalname* is specified, QUERY PROCESS queries all processes belonging to the current *loginname.* If the *terminalname* is specified, the *terminalname* must identify an active terminal. Wildcards may be used in the input parameter specifying the *processID.* If a single wildcard is present (*), it is interpreted as a wildcard for the processID and specifies all processIDs in the system.

QUERY PROCESS returns the following information:

- The *loginname* that owns the process
- The *terminalname* where the login that owns the process is connected
- The *loginID* of the login that the process is in
- The *sessionID* that the process is in
- The process state
- The parent *processID*
- The *processID*
- The process name.

Sample output for QUERY PROCESS follows:



```
  LOGINNAME      TERMNAME LID  SID  STATE PPID   PID
  annm.wrkgrp    serial2  15   1    block 4      35     psel.exe
> annm.wrkgrp    serial2  15   2    ready 4      36     cmd.exe
  miked.wrkgrp   disc     9    1    block 4      18     psel.exe
  miked.wrkgrp   disc     9    2    block 4      27     cmd.exe
  miked.wrkgrp   disc     9    2    exit  27     52     r2eargs.exe
  miked.wrkgrp   disc     9    3    ready 4      51     little.exe
  miked.wrkgrp   disc     9    4    block 4      54     cmd.exe
  admin.wrkgrp   console  7    1    block 4      15     psel.exe
  admin.wrkgrp   console  7    2    block 4      37     cmd.exe
  admin.wrkgrp   console  7    3    block 4      39     cmd.exe
  admin.wrkgrp   console  7    3    ready 39     40     sh.exe
  system.system  disc     1    1    block 0      4      mumproc.exe
  system.system  disc     1    2    block 4      5      hed.exe
  system.system  serial1  16   1    block 4      56     login.exe
```

NOTE: The > character indicates the current process after QUERY PROCESS completes. LOGINNAME identifies the *username* and *groupname* of each user listed. TERMNAME identifies the *terminalname* where the user is logged into the system. LID and SID correspond to the *loginID* and *sessionID*, respectively. STATE indicates the state of the process. The states of a process include the following: blocked (BLOCK), ready (READY), terminating (EXIT), or CPU-starved (STARVE). PPID identifies the parent *processID.* PID identifies the *processID* of the program specified by the following filename.

Security Class Restrictions

QUERY PROCESS may only be used to query processes belonging to the current *loginname*, unless the current *loginname* has sufficient security classification to query processes belonging to other *loginnames.* Administrator class and Operator class have full access to all QUERY PROCESS functions. User class and Guest class may only use QUERY PROCESS functions on processes that belong to the current *loginname.*

Examples

example 1: *processID*

QUERY PROCESS displays information about the specified process with *processID* 8.

```
QUERY PROCESS  8
```

example 2: *sessionID*

QUERY PROCESS displays information about all processes in *sessionID* 3 of login 10 belonging to the *loginname* MIKE.WRKGRP. The following two examples are equivalent.

```
QUERY PROCESS  \MIKE.WRKGRP\10\3

QUERY PROCESS  \MIKE.WRKGRP\10\3\*
```

example 3: *loginID*

QUERY PROCESS displays information about all processes in login 4 belonging to the *loginname* of BRAD.WRKGRP. The following two examples are equivalent.

```
QUERY PROCESS  \BRAD.WRKGRP\4

QUERY PROCESS  \BRAD.WRKGRP\4\*\*
```

example 4: *loginname*

QUERY PROCESS displays information about all processes belonging to the *loginname* of BRAD.WRKGRP. The following two examples are equivalent.

```
QUERY PROCESS  \BRAD.WRKGRP

QUERY PROCESS  \BRAD.WRKGRP\*\*\*
```

example 5: fully specified *processID*

QUERY PROCESS displays information about process 7 in session 3 under login 12 belonging to the *loginname* BRAD.WRKGRP.

```
QUERY PROCESS  \BRAD.WRKGRP\12\3\7
```

example 6: wildcards

QUERY PROCESS displays information about all processes in the system.

```
QUERY PROCESS *
```

example 7: *terminalname*

QUERY PROCESS displays information about all processes running at the terminal PLACE06.

```
QUERY PROCESS \PLACE06
```

QUERY SESSION

Description

Displays information about sessions.

Syntax

QUERY SESSION [*sessionID*] [/V] [/?]

QUERY SESSION
[\ *username.groupname*\ *loginID*\ *sessionID*] [/V] [/?]

QUERY SESSION [\ *terminalname*] [/V] [/?]

Parameters

sessionID

Identifies a session.

\ *username.groupname*\ *loginID*\ *sessionID*

Identifies a specific session belonging to the specific user context (*loginname*, login).

\ *terminalname*

Identifies all sessions running at *terminalname* regardless of user context.

Options

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

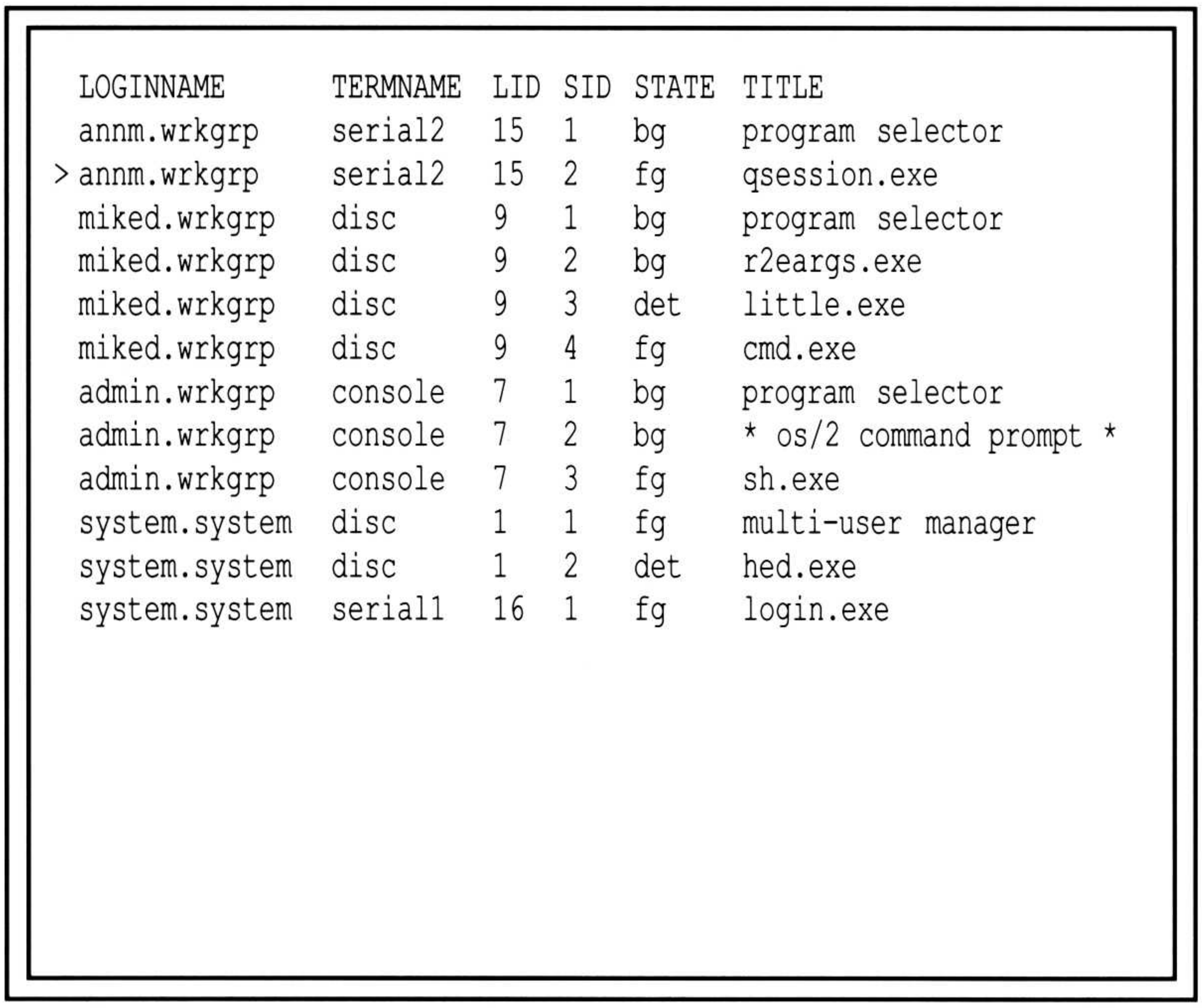
SessionIDs are login-based IDs and are not unique system-wide. In order to uniquely identify a session, it is necessary to specify both the login and the *sessionID.*

If no *sessionID* or *terminalname* is specified, QUERY SESSION displays all sessions in all logins belonging to the current *loginname.* If the *terminalname* is specified, the *terminalname* must identify an active terminal. Wildcards may be used in the input parameter specifying the session. If a single wildcard is present (*), it is interpreted as a wildcard for the *sessionID* and specifies all sessions in the system.

QUERY SESSION returns the following information:

- The *loginname* that owns the session
- The *terminalname* where the login that has the session is connected
- The *loginID* of the login that has the session
- The *sessionID*
- The session state
- The session title.

Sample output for QUERY SESSION is shown below.



```
  LOGINNAME      TERMNAME  LID  SID  STATE  TITLE
  annm.wrkgrp    serial2   15   1    bg     program selector
> annm.wrkgrp    serial2   15   2    fg     qsession.exe
  miked.wrkgrp   disc      9    1    bg     program selector
  miked.wrkgrp   disc      9    2    bg     r2eargs.exe
  miked.wrkgrp   disc      9    3    det    little.exe
  miked.wrkgrp   disc      9    4    fg     cmd.exe
  admin.wrkgrp   console   7    1    bg     program selector
  admin.wrkgrp   console   7    2    bg     * os/2 command prompt *
  admin.wrkgrp   console   7    3    fg     sh.exe
  system.system  disc      1    1    fg     multi-user manager
  system.system  disc      1    2    det    hed.exe
  system.system  serial1   16   1    fg     login.exe
```

NOTE: The > character indicates the current session. LOGINNAME identifies the *username* and *groupname* of each user listed. TERMNAME identifies the *terminalname* where the login is connected. LID and SID correspond to the *loginID* and *sessionID*, respectively. STATE specifies whether the session is foreground (fg), background (bg), or detached (det). TITLE is the title assigned to the session, which defaults to the program *filename* if no title was provided when the session was created.

Security Class Restrictions

QUERY SESSION may only be used to query sessions belonging to the current *loginname*, unless the current *loginname* has sufficient security classification to query sessions of logins belonging to other *loginnames*. Administrator class and Operator class have full access to all QUERY SESSION functions. User class and Guest class may only use QUERY SESSION functions on sessions that belong to the current *loginname*.

Examples

example 1: no options

QUERY SESSION displays information about all the sessions for all logins belonging to the current *loginname*.

```
QUERY SESSION
```

example 2: *sessionID*

QUERY SESSION displays information about sessions with sessionID 3.

```
QUERY SESSION 3
```

example 3: login

QUERY SESSION displays information about all sessions in login 10 belonging to the *loginname* MIKE.WRKGRP. The following two examples are equivalent.

```
QUERY SESSION  \MIKE.WRKGRP\10

QUERY SESSION  \MIKE.WRKGRP\10\*
```

example 4: *loginname*

QUERY SESSION displays information about all sessions for all logins belonging to the *loginname* of BRAD.WRKGRP. The following two examples are equivalent.

```
QUERY SESSION  \BRAD.WRKGRP
```

```
QUERY SESSION  \BRAD.WRKGRP\*\*
```

example 5: *username*

QUERY SESSION displays information about all sessions for the logins belonging to the *loginname* with the *username* BRAD and all *groupnames* for BRAD. The following two examples are equivalent.

```
QUERY SESSION  \BRAD
```

```
QUERY SESSION  \BRAD.*
```

example 6: fully specified *sessionID*

QUERY SESSION displays information about session 2 for login 1 belonging to the *loginname* BRAD.WRKGRP.

```
QUERY SESSION  \BRAD.WRKGRP\1\2
```

example 7: *terminalname*

QUERY SESSION displays information about the sessions for all logins connected at terminal SERIAL0.

```
QUERY SESSION  \SERIAL0
```

example 8: wildcards

QUERY SESSION displays information about all sessions for all logins in the system.

```
QUERY SESSION  *
```

QUERY TERMINAL

Description

Displays information about terminals.

Syntax

QUERY TERMINAL [*terminalname*] [/MODE] [/CONNECT] [/FLOW] [/V] [/?]

Parameter

terminalname
Identifies the terminal.

Options

/MODE (terminal mode)
Displays the current line settings.

/CONNECT (connect settings)
Displays the current connect settings.

/FLOW (flow control)
Displays the current terminal flow control settings.

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

If no *terminalname* is specified, QUERY TERMINAL displays all active terminals in the system. Otherwise, the *terminalname* must identify an active terminal. If a wildcard is present (*), it specifies all active terminals in the system.

Sample output for QUERY TERMINAL follows:

```
  TERMINALNAME    TYPE       DEVICE
  serial3         tvi965     com3         mike's office
  serial1         ibm3151    com2         brad's office
 >serial2         wyse60     com1         ann's office
  console         console    con          administrator's station
```

NOTE: The > character indicates the current terminal. TERMINALNAME specifies the name that is assigned to the terminal. TYPE indicates the terminal type as identified in the terminal profile. DEVICE is the device name that the terminal is assigned. The comment following the terminal information is from the terminal profile.

Security Class Restrictions

Administrator class, Operator class, and User class have full access to QUERY TERMINAL functions. Guest class may only user QUERY TERMINAL to query the current terminal.

Examples

example 1: no options

QUERY TERMINAL displays information about all active terminals.

QUERY TERMINAL

example 2: *terminalname*

QUERY TERMINAL displays information about terminal PLACE07, if it is an active terminal.

QUERY TERMINAL PLACE07

QUERY USER

Description

Displays information about users who are logged into the system.

Syntax

QUERY USER [*username*[*.groupname*]] [/V] [/?]

QUERY USER [\ *terminalname*] [/V] [/?]

Parameters

username[*.groupname*]
Identifies the *loginname*. If the *groupname* is not specified, all *groupnames* for that *username* are used (equivalent to *username.**).

\ *terminalname*
Identifies all *loginnames* with logins at *terminalname.*

Options

/V (verbose)
Displays information about the actions being performed.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

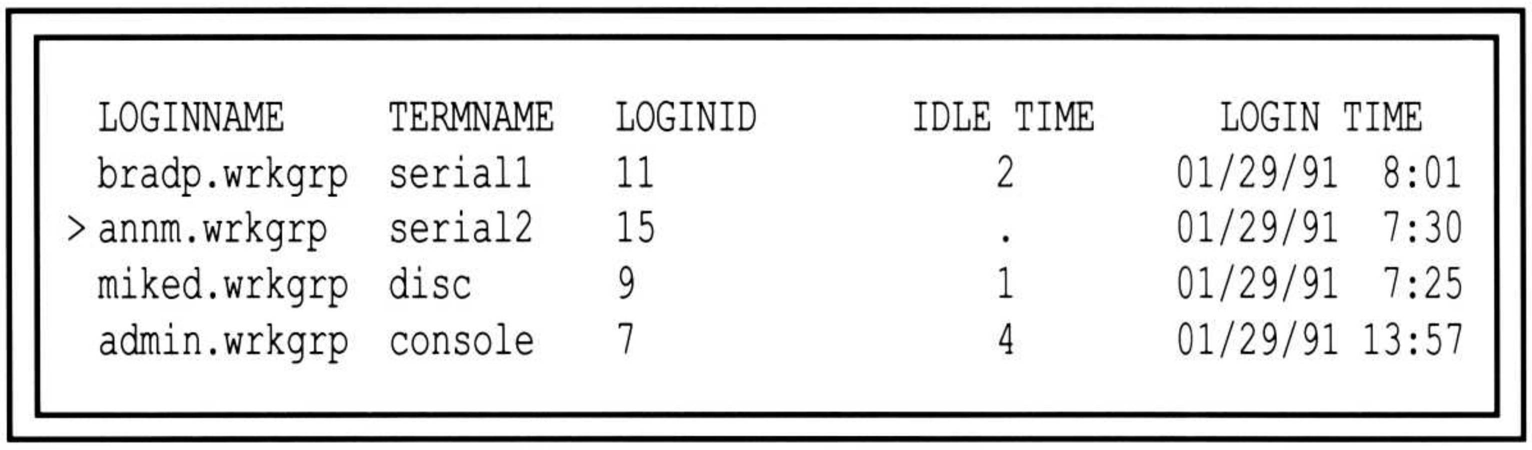
QUERY USER displays information about users (*loginnames*) who are logged into the system. Users who are not logged into the system are not included.

If no *username* or *terminalname* is specified, QUERY USER displays all users who are logged into the system. The *terminalname* must identify an active terminal. Wildcards may be used in the input parameter specifying the *loginname*. If a single wildcard is present (*), it is interpreted as a wildcard for the *username* and specifies all *loginnames* (or users) in the system.

QUERY USER returns the following information:

- The *loginname* of the user
- The *terminalname* where the login is connected.
- The *loginID* of the login
- The idle time, the time since a keystroke was received at the terminal.
- The login time, when the user logged into the system

See the figure below for a sample output of QUERY USER.



```
 LOGINNAME      TERMNAME   LOGINID       IDLE TIME      LOGIN TIME
 bradp.wrkgrp   serial1    11               2        01/29/91  8:01
>annm.wrkgrp    serial2    15               .        01/29/91  7:30
 miked.wrkgrp   disc       9                1        01/29/91  7:25
 admin.wrkgrp   console    7                4        01/29/91 13:57
```

NOTE: The > character indicates the current user. LOGINNAME identifies the *username* and *groupname* of each user listed. TERMNAME identifies the *terminalname* where the user is logged into the system. LOGINID identifies the user login. IDLE TIME is a measure of interaction between the user and the system by indicating the time in minutes since the last keystroke. A dot (.) indicates that the time is less than a minute. LOGIN TIME is the time when the user logged into the system.

Security Class Restrictions

Administrator class, Operator class, and User class have full access to QUERY USER functions. Guest class may only use QUERY USER functions to query the current *loginname.*

Examples

example 1: no options

QUERY USER displays information about all users logged into the system.

```
QUERY USER
```

example 2: *username*

QUERY USER displays information about the *loginname* with *username* BRAD and all *groupname*s associated with it. The following two examples are equivalent.

```
QUERY USER BRAD

QUERY USER BRAD.*
```

example 3: fully specified *loginname*

QUERY USER displays information about the *loginname* BRAD.WRKGRP.

```
QUERY USER BRAD.WRKGRP
```

example 4: *terminalname*

QUERY USER displays information about all *loginnames* with logins at terminal TERM08.

```
QUERY USER \TERM08
```

example 5: wildcards

In the first case, QUERY USER displays information about all active *loginnames* with the *groupname* of WRKGRP. In the second case, QUERY USER displays information about all active *loginnames* in the system.

```
QUERY USER *.WRKGRP
QUERY USER *
```

RECOVER (Recover Files)

Description

Reconstructs all the files from a disk that has a damaged directory structure or reconstructs a single file from a disk that has bad sectors.

Syntax

RECOVER [*drive*:] | [*drive*:][*path*]*filename*

Parameters

drive:

Specifies the *drive* containing the disk with the files you want to try to reconstruct. If you specify only a *drive*, the RECOVER utility tries to reconstruct the entire contents of the disk in that drive, giving the reconstructed files names of the form *FILEnnnn.REC*, where *nnnn* is a four-digit number. In this case, the files being reconstructed lose their extended attributes. The RECOVER utility reconstructs the extended attributes, if possible, and stores them in a separate *FILEnnnn.REC* file.

filename

Specifies the file you want to try to reconstruct. If you do not specify a *drive* or *path*, the RECOVER utility uses the current directory on the current drive. You cannot use wildcard characters in *filename*. If the file has extended attributes, RECOVER tries to reconstruct all of them along with the file, but if it cannot do so, it puts whatever extended attributes it can reconstruct into a file named EA0001.REC.

Remarks

MS OS/2 *MULTIUSER* recovers a file by reading it, sector by sector, skipping the bad sectors.

All files created by the RECOVER utility are stored in the root directory of the specified drive.

When you use the RECOVER utility on a disk or partition that is formatted for HPFS, you can reconstruct only one file at a time. The utility reconstructs the file with its original *filename* and location, if possible.

If you type RECOVER by itself (in the FAT file system), the utility tries to reconstruct all the files in the current directory on the current drive.

For more detailed information on the RECOVER command, see Chapter 13, "System Maintenance" in the *Citrix MULTIUSER System Administrator's Guide.*

WARNING: Before you use this utility, make a backup copy of your disk and try to restore the files by using the RESTORE utility. If this operation fails, use the RECOVER utility to try to reconstruct the lost files one by one. Do not use RECOVER to reconstruct an entire disk unless the entire disk is unreadable.

You cannot use the RECOVER utility on the drive from which you started your system, nor on the drive from which you are running RECOVER. To reconstruct files on your start-up drive, you must run RECOVER from the MS OS/2 *MULTIUSER* Install disk. You can do the same thing to reconstruct files on the drive from which you would normally run RECOVER, or you can simply copy the utility to another drive and run it from there.

Security Class Restrictions

Administrator and Operator classes have permission to use the RECOVER utility on fixed disk partitions. Any user can execute RECOVER on diskettes. An error message is displayed if a user who is not authorized to do so attempts to use the RECOVER utility.

Example

To try to reconstruct the file *REPORT.TXT* from the disk with bad sectors in drive B, type the following:

```
RECOVER  B:\REPORT.TXT
```

To try to reconstruct the file *JANUARY.FIL* from drive C, which is your start-up drive, SHUTDOWN the system. Then, insert the MS OS/2 *MULTIUSER* Install disk in drive A, press **CTRL+ALT+DEL**, press **ESC** when the logo appears, and then type the following at the command prompt:

```
RECOVER  C:\JANUARY.FIL
```

REGISTER (Secure a Program)

Description

This utility is used to register a program so that it can have special security and execution characteristics.

Some complex applications and subsystems need to execute in a "system global" context so that they can establish resources that are common to more than one user. It is necessary to register these types of applications.

In addition, some applications need special security authorizations to perform system functions not available to programs in general. These also need to be registered.

Syntax

REGISTER [*pathname*] [/ALIAS:*aliasname*] [/Q] [/D] [/SYSTEM] [/USER] [/?]

Parameter

pathname

This is the program *filename* or *pathname* being registered or queried. You can register a specific file (*.EXE* or *.DLL*), or you can register a directory path. This parameter cannot contain wildcards.

Options

/ALIAS:*aliasname*

This specifies an alias under which the *pathname* is registered.

/Q

Queries current registration status. If /ALIAS is specified, this displays all of the programs registered under the specified ALIAS. If *pathname* is given, this displays the ALIAS under which the program is registered.

/D

Deletes the registration for the *pathname* given. If /ALIAS is specified and *pathname* is not specified, all programs registered under the *aliasname* are de-registered.

/SYSTEM

Specifies that *pathname* is system global.

/USER

Specifies that *pathname* is user global. This is the default when a program is being registered.

/? (help)

Displays the syntax for the utility and information about the utility's options.

Remarks

Multiple programs and program paths can be registered under the same alias. The result is that all programs under that alias are treated the same when that *aliasname* shows up in a security attribute. Refer to Chapter 7, "Configuring Security" in the *Citrix MULTIUSER System Administrator's Guide.*

The *pathname* parameter must give either a filename or a directory name. Incomplete paths are allowed (such as ..\XYZ.EXE). REGISTER always fully qualifies the path prior to registration. Thus the following commands

```
CD \USR\ME
REGISTER MYPROG.EXE /ALIAS:NEWPROGRAMS /SYSTEM
```

will register C:\USR\ME\MYPROG.EXE under the *aliasname* "newprograms." Only that program, run from that explicit path, will be registered; executing a different copy of MYPROG from a different path will not execute the registered version.

The *pathname* parameter can also give a directory path. This is a simplified way to register a group of programs under the same *aliasname.* If a program within a registered path is also registered under a different *aliasname*, the explicit register of the program takes precedence.

If a program or program path is already registered and a new REGISTER command is given for this program, the program is "re-registered" under the new alias.

You can change the context of a program that is already registered without entering the ALIAS parameter.

Registration data for a program is recognized only when the program is loaded. Therefore, if you issue a REGISTER command for a program that is already loaded, the changes will not take effect until the next time the program is loaded.

Security Class Restrictions

This utility is available only to an Administrator or Operator class.

Examples

REGISTER \APP\DLLS /ALIAS:SpecialPrograms

Registers the program path \APP\DLLS under the alias name "SpecialPrograms". All programs (*.EXE* and *.DLL*) in that path will have the security characteristics associated with the name SpecialPrograms. User global context is assumed (by default).

REGISTER SUBSYS.DLL /SYSTEM

Sets the execution context of *SUBSYS.DLL* to system global. In this example *SUBSYS.DLL* has already been registered.

REGISTER /ALIAS:SpecialPrograms /Q

Displays all programs and paths registered under the name SpecialPrograms.

REGISTER SUBSYS.DLL /D

De-registers *SUBSYS.DLL.*

REM (Remark)

Description

Includes a descriptive comment in a batch file.

Syntax

REM [*comment*]

Parameter

comment

Specifies the message you want to put in the batch file. *comment* can be any combination of characters that fits on one line.

Remarks

If the echo feature is on when MS OS/2 *MULTIUSER* encounters a REM line in a batch file, the system displays the line; if the echo feature is off, MS OS/2 *MULTIUSER* does not display the line.

If the *comment* you want to put in the batch file is too long to fit on one line, you must use the REM command again for each line in the comment. You can also use REM without comment to add spacing between blocks of comments or remarks.

Example

The following batch file uses the REM command both to include comments in the file and to add spacing:

```
@echo off
REM This batch file displays the directory
REM of the disk in drive A.
REM
REM You use the /W option to display
REM the directory in wide format.
REM
DIR A: /W
```

RENAME (Rename File)

Description

Changes the name of a file or directory. You can abbreviate RENAME as REN.

Syntax

RENAME [*drive*:][*path*]*filename1 filename2*

Parameters

filename1
Specifies the old name of the file or directory. If you do not specify a drive and path, MS OS/2 *MULTIUSER* uses the current directory on the current drive.

filename2
Specifies the new name of the file or directory. You cannot specify a drive or path for the new name; the RENAME command changes only the name of the file or directory.

Remarks

You can use wildcard characters in *filename1* and *filename2.*

You cannot use the RENAME command to move files from one drive or directory to another.

If a file or directory with the name specified for *filename2* already exists in the parent directory, MS OS/2 *MULTIUSER* displays an error message.

You must have Create (C) permission on the file being renamed.

NOTE: In the High-Performance File System (HPFS), you can use the RENAME command to change the case of a *filename* or directory name by renaming the file or directory as itself. For example, you can change the filename *"mytaxfile.txt"* to *"MYTAXFILE.TXT"* by using the uppercase version of the name as the *filename2* argument. For more information about HPFS, see Chapter 3 of the *Citrix MULTIUSER User's Guide.*

Example

To change the extension of all the files in the current directory in drive B that have a .DOC extension from *.DOC* to *.TXT*, type the following:

```
RENAME B:*.DOC  *.TXT
```

REPLACE (Replace Files)

Description

Selectively replaces files in the destination directory with new versions of those files from the source directory, or adds new files to the destination directory.

Syntax

REPLACE [*drive*:][*path*]*filename* [[*drive*:]*path*]
[[/A | /S] | [/A | /U]] [/P] [/R] [/W] [/F]

Parameters

filename
: Specifies the source file that is to replace the file on the destination disk. You can use wildcard characters in the source *filename* to replace groups of files that have similar names.

path
: Specifies the directory that contains the file to be replaced or to which you want to add the file. If you do not specify a directory, REPLACE uses the current directory on the current drive.

Options

/A

Adds only the files specified in *filename* that do not exist in the destination directory but does not replace existing files. You cannot use this option with the /S option.

/P

Prompts you for confirmation before replacing or adding a file.

/R

Replaces read-only files as well as unprotected files. If you do not use this option and you try to replace a read-only file, the replacement process stops and you see an error message.

/S

Replaces files in the subdirectories of the destination directory if the *filenames* match those specified in *filename*. The REPLACE utility never searches subdirectories of the source directory. You cannot use this option with the /A option.

/U

Replaces only the target files that are older than their corresponding source files. You cannot use this option with the /A option.

/W

Instructs the REPLACE utility to wait for you to insert a disk before beginning to search for source files. Otherwise, REPLACE starts replacing or adding files immediately.

/F

Specifies that the REPLACE utility should not discard the extended attributes of a file if the destination file system does not support extended attributes. In this case, the utility does not replace the file.

Remarks

The REPLACE utility is usually used to update the software on your fixed disk. You cannot use REPLACE to update hidden files or system files.

The REPLACE utility copies the extended attributes of the new version of the file (the source file) to the destination file. The REPLACE utility does not copy security attributes.

If you omit the /F option when you use the REPLACE utility to copy a file with extended attributes to a file system that does not support extended attributes, the utility copies the file and discards the extended attributes. If the file requires the extended attributes, REPLACE does not copy the file but displays an explanatory message on the screen.

You must have Read (R) permission on the source file and Create/Write (C/W) permission on the destination file.

Examples

To update your phone-list file, *PHONES.CLI*, in all the directories on your fixed disk (drive C) with the latest version of *PHONES.CLI* from the disk in drive B, type the following:

```
REPLACE  B:\PHONES.CLI  C:\ /S
```

To add several new printer drivers from the disk in drive A to the directory MSTOOLS on your fixed disk (drive C), type the following:

```
REPLACE A:*.PRD C:\MSTOOLS /A
```

RESERVE (Claim a Device)

Description

This utility is used to RESERVE a device for exclusive use. This prevents another user from reserving and gaining access to the device. This utility is also used to release a device reservation.

Syntax

RESERVE *device*: [/Q] [/D] [/?]

Parameter

device:
Specifies the device name.

Options

/Q
Queries current reservation status of the device.

/D
Deletes the current reservation of the device.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

This command does not attempt to open the device, nor does it make any attempt to verify that the device exists. Its purpose is to temporarily change the security characteristics of the device such that only the user reserving the device can successfully access (open).

The device must be set up as one that can be serialized (RESERVE command) by the Administrator, otherwise the request is rejected. If the device is set up in this way, you cannot use the device until you successfully reserve it. Diskette drives A and B generally require RESERVE.

Once a device is reserved, it remains reserved until it is released (using /D) or until the last instance of the loginname who has reserved the device logs out.

Security Class Restrictions

You may not reserve a device unless you have Reserve (V) permission for that device.

You may not delete (/D) a device reservation unless you have the device reserved.

Administrators can delete a reservation on a device regardless of what user has it reserved.

Examples

RESERVE A:
    Reserves drive A for exclusive use.

RESERVE A: /D
    Removes the reservation from drive A so that other users may reserve the drive.

RESET (Restore State)

Description

The RESET utilities are used to reset current settings to the values that are stored in the profiles.

Syntax

```
RESET [AUDIT | EVENTS | LIMITS | TERMINAL]  [/?]
```

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

The RESET utility will undo any changes made by the CHANGE utility. Resetting a terminal causes the terminal to restart using the current terminal profile. Any logins on the terminal are ended.

RESET AUDIT

Description

Resets the system-wide security audit events that are currently being logged to be equal to those specified in the System Profile.

Syntax

RESET AUDIT [/?]

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

The audited events are established using CONFIG SYSTEM and are placed in the system profile. When the system is started, the current audit logging state is taken from the system profile. At any time during system operation, the selection of audit events being logged can be changed using CHANGE AUDIT. The RESET AUDIT command resets the state of the auditing to that specified in the system profile.

This only resets the system-wide auditing flags. Audit flags can also be placed on users and resources using CONFIG USER and CONFIG ACCESS. This command does not affect the user and resource audit flags.

Auditing can be turned on and off by using CHANGE EVENTS. If auditing is off, the audit flags established here will not be used until audit event logging is turned on.

Security Class Restrictions

This utility is available only to an Operator or Administrator.

RESET EVENTS

Description

Resets the events that are currently being logged to be equal to those specified in the System Profile.

Syntax

```
RESET EVENTS [/?]
```

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

The logged events are established using CONFIG SYSTEM and are placed in the system profile. When the system is started, the current event logging state is taken from the system profile. At any time during system operation, the selection of events being logged can be changed using CHANGE EVENTS. The RESET EVENTS command resets the state of the event logging to that specified in the system profile.

Security Class Restrictions

This utility is available only to an Operator or Administrator.

RESET LIMITS

Description

RESET LIMITS resets user resource limits to their initial login values, found in the user's profile.

Syntax

RESET LIMITS [*username*[.*groupname*]] [/MIN*resource*] [/MAX*resource*] [/?]

Parameter

username[.*groupname*]
: Identifies the *loginname* of the user whose resource limits are to be reset. If the *groupname* is not specified, all groupnames for that *username* are used (equivalent to *username.**).

Options

/MIN*resource* and /MAX*resource*
: Specifies what resource minimum or maximum limit is to be reset. More than one of these can be specified. If no *options* are specified, then all the possible resource limits are reset.

The following specific resource limit options are available for /MIN*resource* and /MAX*resource*:

/MINMEMORY
/MAXMEMORY
/MINTHREADS
/MAXTHREADS
/MINFILES
/MAXFILES
/MINSEMAPHORES
/MAXSEMAPHORES
/MAXLOGINS
/MAXSESSIONS

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

The default for the RESET LIMITS command is to reset all of the resource limits for the current *loginname*. If RESOURCE=OFF in *CONFIG.SYS*, the message "Resource Management is turned off" is displayed.

Security Class Restrictions

Only an Operator or an Administrator can reset any user's resource limits.

Examples

The *loginname* can contain wildcard characters as in the following examples, provided the user has the proper security classification:

```
RESET LIMITS *.*
RESET LIMITS *.WRKGRP
RESET LIMITS MIKEH.WRKGRP /MAXLOGINS
```

RESET TERMINAL

Description

RESET TERMINAL resets the terminal subsystem hardware and software to known initial values.

Syntax

RESET TERMINAL *terminalname* [/?]

Parameter

terminalname
The *terminalname* is the name given to the terminal by the System Administrator when the terminal was configured. QUERY TERMINAL can be used to determine the *terminalname*.

Option

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

RESET TERMINAL causes the device to be closed and then reopened. If the terminal exists but is in the disabled state, the terminal is enabled as well as reset. If the terminal configuration exists but the terminal is not functional, the terminal is made functional.

RESET TERMINAL is used when the terminal is malfunctioning or the terminal appears to be down. The terminal is reset, the underlying hardware is reset, the terminal drivers and terminal capability data are reloaded and reinitialized.

RESET TERMINAL is also used to take the terminal out of the disabled initial state. There is a configuration parameter which allows for the terminal to be configured to start in the disabled state. Again, although the terminal is started by the RESET TERMINAL utility when the terminal is in the disabled initial state, the permanent configuration record is not changed by executing the RESET TERMINAL function. After the terminal is started, the terminal becomes disabled again when the terminal connection is broken, when the terminal is powered off, or when the last user on the terminal logs out. CONFIG TERMINAL is used to change the permanent configuration record.

Finally, RESET TERMINAL is used to allow a terminal to be configured without having the terminal or the terminal cables available. An attempt is made during the running of CONFIG TERMINAL to bring the terminal up. If this fails, CONFIG TERMINAL proceeds as normal. When the terminal is finally physically connected, the RESET TERMINAL is run.

Security Class Restrictions

An Administrator or Operator can use the RESET TERMINAL command to reset any terminal in the system.

A User or Guest may only use the RESET TERMINAL command to reset any terminal on which he or she is logged into.

RESTORE (Restore Saved Files)

Description

Restores backup files that were created by using the BACKUP utility.

Syntax

RESTORE *drive1*: *drive2*:[*path*][*filename*] [/S] [/P] [/B:*date*] [/A:*date*] [/E:*time*] [/L:*time*] [/M] [/N] [/F]

Parameters

drive1:
Specifies the drive that contains the disk on which the backup copies of your files are stored.

drive2:
Specifies the drive that contains the destination disk to which you want to restore the files. The destination disk does not have to be of the same type as the source disk.

filename

Specifies the file that you want to restore. You can use wildcard characters to specify multiple files with similar names. If you do not specify a *filename*, the RESTORE utility searches the current directory on the current drive for *filenames* that match those found on *drive1*.

Options

/S

Restores the files in subdirectories of the directory that contains *filename*.

/P

Prompts you for confirmation before restoring read-only files or files that have changed since the last backup operation.

/B:*date*

Restores only files that were modified on or before the specified *date*.

/A:*date*

Restores only files that were modified on or after the specified *date*.

/E:*time*

Restores only files that were modified at or before the specified *time*. Do not use this option without the /B:*date* or /A:*date* option.

/L:*time*

Restores only files that were modified at or after the specified *time*. Do not use this option without the /B:*date* or /A:*date* option.

/M

Restores only files that have been modified since the last backup operation and turns off the archive bits of the restored files.

/N

Restores only files that no longer exist on the destination disk.

/F

Specifies that the RESTORE utility should not discard the extended attributes of a file if the destination file system does not support extended attributes. In this case, the utility does not restore the file.

Remarks

The RESTORE utility can restore files only to their original directory.

The RESTORE utility cannot restore *CMD.EXE, COMMAND.COM,* or the hidden system files, nor can it restore files that are in use on your start-up drive. To restore the backup copies of the files on your start-up drive, you must run RESTORE from the MS OS/2 *MULTIUSER* Install disk.

The RESTORE utility restores backup files that were created by using the MS OS/2 *MULTIUSER* BACKUP utility or the MS-DOS BACKUP utility (including versions 3.21 and earlier of MS-DOS, even though they use a different structure for backup files).

The RESTORE utility preserves the extended attributes and security attributes of a file or directory.

If you omit the /F option when you use the RESTORE utility to copy a file with extended attributes to a file system that does not support extended attributes, the utility copies the file and discards the extended attributes. If the file requires the extended attributes, RESTORE does not copy the file but displays an explanatory message on the screen.

RESTORE can be used to restore system files; however, you may need to be in maintenance mode to complete this task. See Chapter 13, "System Maintenance" in the *Citrix MULTIUSER System Administrator's Guide* for instruction on restoring system files.

You must have Create (C) and Write (W) permissions to RESTORE files to a directory.

Example

To restore all the backup files on drive A with the extension *.NEW* to the directory LETTERS on the disk in drive B, type the following:

```
RESTORE A: B:\LETTERS\*.NEW
```

RMDIR (Delete Directory)

Description

Deletes a directory. You can abbreviate RMDIR as RD.

Syntax

RMDIR [*drive*:]*path* [...]

Parameter

path
Specifies the directory that you want to delete.

Remarks

Before you can delete a directory, you must delete all the files and subdirectories in that directory. (The empty directory still shows listings for the working directory [.] and the parent directory [..] if you list its contents with the DIR command.)

You can delete more than one directory at a time by specifying each directory separately on the same command line.

To use the RMDIR command, you must have Delete (D) permission to the directory.

Example

To remove the directory BILLS and the directory NOTICES from the root directory on drive B, type the following:

```
RMDIR B:\BILLS B:\NOTICES
```

SET (Set Environment)

Description

Defines an environment variable by naming the variable and giving a value for it.

Syntax

SET [*string1*=[*string2*]]

Parameters

string1
  Specifies the name of the environment variable you want to set (for example, PATH, INIT, LIB, or PROMPT).

string2
  Specifies the *string* of characters, paths, or *filenames* that defines the current value of the environment variable.

Remarks

If you type SET by itself, MS OS/2 *MULTIUSER* displays the current values for all the environment variables in the current session.

If you type SET STRING1=, MS OS/2 *MULTIUSER* removes the current value for that environment variable.

For more information about environment variables, see Chapter 3 in the *Citrix MULTIUSER User's Guide.*

You can also use the SET command to define the replaceable batch parameters by name, instead of by number. For more information about using the SET command in batch files and start-up files, see Chapter 6 in the *Citrix MULTIUSER User's Guide.*

If used from the command prompt, the SET command affects only the current session.

Example

To set the INCLUDE environment variable so that the Microsoft C Optimizing Compiler can find INCLUDE files in the directory INC on drive C, type the following:

```
SET INCLUDE=C:\INC
```

SETLOCAL (Set Local Environment)

Description

Saves the current drive, directory, and environment settings and specifies to MS OS/2 *MULTIUSER* that any changes to these settings that you make in a batch file are valid only while the batch file is running. Use this command only in a batch file.

Syntax

```
SETLOCAL
```

Remarks

After typing SETLOCAL on a line by itself, you can set any variables you want for the purposes of the batch file; MS OS/2 *MULTIUSER* restores the original settings when it encounters an ENDLOCAL command or when the batch file ends.

You can use multiple SETLOCAL commands in a batch file without including corresponding ENDLOCAL commands; each succeeding SETLOCAL overrides the previous one, in effect acting as an ENDLOCAL command for the previous SETLOCAL.

Example

To have MS OS/2 *MULTIUSER* search the root directory of the disk in drive B for programs, even though your normal path does not include drive B, type the following in the batch file:

```
SETLOCAL
PATH B:
  .
  .
  .
ENDLOCAL
```

SHIFT (Additional Batch Parameters)

Description

Shifts the positions of the numbered replaceable parameters (%0-%9) in a batch file (*.CMD* extension), so that you can use more than 10 values. Use this command only in a batch file.

Syntax

```
SHIFT
```

Remarks

Each time MS OS/2 *MULTIUSER* encounters the SHIFT command, it discards the value in %0 and shifts each of the other values to the next lowest replaceable parameter: the value in %1 moves to the %0 position, %2 moves to %1, and so on. This frees the last parameter, %9, so that the next value you type shifts into %9.

You can use the SHIFT command as many times as necessary in a batch file. You can use SHIFT even if there are fewer than 10 values.

There is no backward SHIFT command. After you have used shift once, the value in the original %0 is gone and the batch file cannot use it again.

Example

The following batch file (named *COPYTO.CMD*) uses the SHIFT command to copy a list of files to your current directory:

```
@ECHO OFF
IF "%1"=="" GOTO USAGE
:START
COPY %1
SHIFT
IF "%1"=="" GOTO DONE
GOTO START
:USAGE
ECHO * THIS BATCH PROGRAM COPIES ANY NUMBER
ECHO * OF FILES TO YOUR CURRENT DIRECTORY.
ECHO *
ECHO * USAGE: COPYTO FILE1 FILE2 FILE3 [...]
:DONE
```

SHUTDOWN (Shut Off the System)

Description

Quiesces the system by terminating all logins, closes the file system, and shuts down the system.

Syntax

SHUTDOWN [/F] [/MAINT] [/REBOOT] [/NOMSG] [/V] [/?] [+*minutes*]

Options

/F (force)
: Forces system shutdown to occur. Otherwise, prompts to ensure that a shutdown is desired at this time.

/MAINT (maintenance mode)
: Enters maintenance mode after all logins have terminated.

In maintenance mode, all terminals except the terminal from which the SHUTDOWN command was issued will be disabled. The MS OS/2 *MULTIUSER* command interpreter, CMD, will be started at this terminal. For more information about maintenance mode, refer to Chapter 13, "System Maintenance" in the *Citrix MULTIUSER System Administrator's Guide*.

/REBOOT (restart)
: Closes the file system (HPFS) drives and restarts the system.

/NOMSG

Disables the sending of a message to all users notifying them that the system is shutting down.

/V (verbose)

Displays information about the actions being performed.

/? (help)

Displays the syntax for the utility and information about the utility's options.

+*minutes*

Number of minutes from the current time until the system will be shutdown.

Remarks

SHUTDOWN performs the following tasks, allowing you to power off the system, if desired.

- Sends a message to all users notifying them of the shutdown (unless the /NOMSG option is specified).
- Terminates all active logins
- Closes the file system
- Halts the system

It is important to use the SHUTDOWN utility to halt the system when you are using the High-Performance File System (HPFS). SHUTDOWN closes the file system in an orderly fashion and ensures that the file system is in a consistent state.

The /MAINT and /REBOOT options are mutually exclusive. You may not specify both options at the same time.

After using SHUTDOWN with the /MAINT option to enter maintenance mode, and after performing maintenance activities, you may use SHUTDOWN with the /REBOOT option to restart the system in normal operating mode.

Security Class Restrictions

Use of SHUTDOWN is restricted to those *loginnames* with sufficient security classification to affect system operations. Only Administrator class and Operator class have access to SHUTDOWN functions.

Examples

example 1: no options

SHUTDOWN prompts you to determine if system shutdown should occur.

```
SHUTDOWN
```

example 2: reboot option

SHUTDOWN terminates all logins, closes the file system, and restarts the system. No prompt occurs because of the /F option.

```
SHUTDOWN /F /REBOOT
```

example 3: maintenance mode option

SHUTDOWN terminates all logins and causes the system to enter maintenance mode. The MS OS/2 *MULTIUSER* command interpreter, CMD, is started at the terminal from which SHUTDOWN was issued. No prompt occurs because of the /F option.

```
SHUTDOWN /F /MAINT
```

example 4: halt the system in 5 minutes

SHUTDOWN notifies all users that a system shutdown is in 5 minutes, then 5 minutes later halts the system. No prompt occurs because of the /F option.

```
SHUTDOWN /F +5
```

SORT (Sort Data)

Description

Sorts lines of input in alphabetical and numeric order and sends the output lines to the screen.

Syntax

SORT [/R] [/+N] < *source*

Options

/R

Sorts the lines in reverse order (that is, from Z to A, then from 9 to 0).

/+N

Starts sorting the lines according to the character in column n (that is, the nth character from the beginning of the line). If you do not specify this option, the SORT utility starts sorting according to the first character of each line.

Parameter

source

Specifies the *source* of the input. The keyboard is the default source, but you can also redirect a file or the output from another command. For more information about redirection, see Chapter 3, "Redirecting Input, Output, and Error Messages" in the *Citrix MULTIUSER User's Guide.*

Remarks

The SORT utility uses the collating sequence table that is appropriate to the country-code and code-page settings. It does not distinguish between uppercase and lowercase letters.

NOTE: If you try to SORT a file that is too large, the SORT utility displays an error message. A file is too large if it is larger than 63K or if the number of lines is greater than the value ((file size in bytes + 768) / 4). To sort a large file, split the file into two or more smaller files.

Examples

To sort the contents of the file PHONE.TXT in reverse alphabetical order and display the output on your screen, type the following:

```
SORT /R < PHONE.TXT
```

To sort a listing of the filenames in the CLIENTS directory on drive B and send the sorted list to the file CLIENTS.LST on drive C, type the following:

```
DIR B:\CLIENTS | SORT > C:CLIENTS.LST
```

SPOOL (Control Multiuser Printing)

Description

Controls the operations of the spooler. It can:

- Redirect printing output from one device, such as a parallel port, to a different output printing device.
- Change status of queues and jobs.
- Query status of queues and jobs.

Syntax

SPOOL [*username*[*.groupname*]] [/Q] [/D:*device1*] [/O:*device2*] [/HQ:*queuename*] [/RQ:*queuename*] [/CQ:*queuename*] [/DQ:*queuename*] [/QD:*queuename*] [/CJ:*jobID*] [/PN:*jobID*] [/PA:*jobID*] [/RJ:*jobID*] [/HJ:*jobID*] [/SP:*pathname*] [/?]

Parameter

username[.*groupname*]

Display the print queues and all print jobs in the print queues for the *loginname.* If the *groupname* is not specified, all groupnames for that *username* are used (equivalent to username.*), see the /Q option.

Options

/Q

Displays the print queues and all print jobs in the print queues for the system. Note that SPOOL /Q is equivalent to SPOOL *. If you do specify *loginname*, this option is ignored.

/D:*device1*

Identifies the parallel port that is connected to the printing device. The default device is LPT1. You cannot specify a serial device as *device1*, but you can use any parallel device (LPT1, LPT2, and so forth) that supports monitors.

/O:*device2*

Identifies the output printing device. You can specify any parallel port (LPT1, LPT2, and so forth, or PRN) or any serial port (COM1, COM2, and so forth). If you do not specify *device2*, the SPOOL utility uses *device1* as the default output printing device.

/HQ:*queuename*

Holds the queue specified in *queuename.* To hold a queue means to prevent it from sending print jobs that are in it to a printer. If a job is currently printing, it is not affected by the hold. Holding a queue that is already held has no effect.

/RQ:*queuename*

Releases the queue specified in *queuename.* To release a queue means to return it to an active state. Releasing a queue that is not held has no effect.

/CQ:*queuename*

Cancels all jobs in the queue.

/DQ:*queuename*

This will remove the queue. If there are jobs in the queue, they are printed and the queue is removed.

/QD:*queuename*

This will display the details of the specified queue.

/CJ:*jobID*

Cancels the print job and removes it from the queue.

/PN:*jobID*

Prints job next. The job is moved to the head of the queue. If the job is held, it is released and moved to the head of the queue.

/PA:*jobID*

Prints job again. This will stop the job from printing and start printing it again from the beginning. If the job is not printing, this has no effect.

/RJ:*jobID*
Releases a print job. A job that is released is eligible for printing again. Releasing a job that is not held has no effect.

/HJ:*jobID*
Holds a print job. A job that is held will not be printed until it is released. A job that is currently printing is not held.

/SP:*pathname*
This will allow you to specify the spool path. You must type in the full *pathname.*

/? (help)
Displays the syntax for the utility and information about the utility's options.

Remarks

SPOOL with no parameters will start the full screen utility. Refer to Chapter 8 and Chapter 15 of the *Citrix MULTIUSER System Administrator's Guide* and Chapter 4 of the *Citrix MULTIUSER User's Guide* for more information on the SPOOL full screen utility.

The spooler intercepts files that are being sent by the PRINT utility or an application to a printer, holds the files temporarily in a directory on disk, and then prints them one at a time.

Because the output printing device can be different from the one specified in the printing command, you can use the SPOOL utility to redirect the parallel printer output (for example, to an asynchronous serial device).

Security Class Restrictions

Administrators and Operators can query and modify all print jobs and all print queues. Users and Guests may view but not modify print queues. A User may query all print jobs and can modify only their own print jobs. A Guest may only query and modify their own print jobs.

A User and Guest class may not use the /PN:*jobID* (Print Job Next) option.

Examples

To spool your print jobs and print them on a serial printer connected to COM1, type the following:

```
SPOOL /O:COM1
```

To hold all jobs in queue "LPT1Q", type the following:

```
SPOOL /HQ:LPT1Q
```

START (Start Another Program in a New Session)

Description

Starts an MS OS/2 *MULTIUSER* command interpreter and tells it to carry out the command you specify.

Syntax

START ["*session*"] [/K | /C | /N] [/F] [/I] [/PGM] [*drive*:][*path*] *command*[.*ext*] [*arguments*]

or

START "*session*" [/K | /C] [/F] [/I] "[*drive*:][*path*] *command*[.*ext*] [*arguments*]"

or

START ["*session*"] [/K] [/F] [/I]

Parameters

session
Specifies the name of the new *session* as it will appear in the Task List. The name can be up to 60 characters and can include spaces; it must be surrounded by double quotation marks. If you do not specify a name, MS OS/2 *MULTIUSER* uses the *filename* you specify for the command argument. If you do not specify either the *session* argument or the *command* argument, MS OS/2 *MULTIUSER* uses *CMD.EXE*.

command[.*ext*]
Specifies the command you want the new command interpreter to carry out. This may be an MS OS/2 *MULTIUSER* command, a utility, a batch file, or a command that starts an application. If you do not specify an extension, MS OS/2 *MULTIUSER* searches for command with the extensions *.COM*, *.EXE*, and *.CMD*, in that order.

Options

/K

Instructs the new command interpreter to run the *command* you specify and then keep the *session* open when the command is completed. This is the default option.

/C

Instructs the new command interpreter to carry out *command* and then end the *session* and return to the program from which it was started.

/N

Instructs MS OS/2 *MULTIUSER* to run *command* without starting a new command interpreter. You cannot use /N if you have enclosed the *command* and its *arguments* in double quotation marks (see the following *arguments* description), nor can you use /N if *command* is a batch file or attempts to use a batch file (batch files require *command* as their batch processor).

/F

Instructs MS OS/2 *MULTIUSER* to run *command* in the foreground. Note that if you use several start commands in a batch file, you can use only one /F option; MS OS/2 *MULTIUSER* ignores all but the last one.

/I

Instructs MS OS/2 *MULTIUSER* to give the new session the default environment which is derived from the environment specified in the *CONFIG.SYS* file. The environment includes environment variables such as PATH, DPATH, and the drive and directory for a session.

/PGM
Specifies that the quoted *string* following this option in the command line is the name of the program to be run.

arguments
Specifies any valid *arguments* for *command.*

Remarks

If you type START without specifying a command, MS OS/2 *MULTIUSER* starts a new command interpreter (CMD) without running a command.

If you specify a quoted string in the START command, MS OS/2 *MULTIUSER* interprets the quoted string as the session name unless you use the /PGM option with the quoted string.

You must have Execute (X) permission for the program being started.

Example

To start a new MS OS/2 *MULTIUSER* command interpreter and run an application titled Videophile in the foreground session, type the following (assuming that the command you use to start Videophile is videop):

```
START "Videophile" /F videop
```

SYSLOG (Log File Control)

Description

Suspends or resumes system-event logging. When logging is on, MS OS/2 *MULTIUSER* records system events in the system log file.

Syntax

```
SYSLOG [/S | /R]
```

Options

/S
Suspends system-event logging.

/R
Resumes system-event logging.

Remarks

If you type SYSLOG by itself, MS OS/2 *MULTIUSER* starts the utility and displays its main menu. You can use the /S and /R options if you have specified the log command in your *CONFIG.SYS* file.

The SYSLOG utility should not normally be used. Instead, logging should be controlled with the EVENTS utility. For a discussion of event logging, see Chapter 14 of the *Citrix MULTIUSER System Administrator's Guide.*

Security Class Restrictions

This utility can be executed only by the Administrator or Operator class.

TIME (Set or Display Time)

Description

Queries and sets the system clock by specifying the current time.

Syntax

TIME [*hours*:*minutes*][:*seconds*[.*hundredths*]]

Parameters

hours
Specifies a number in the range 0 through 23.

minutes
Specifies a number in the range 0 through 59.

seconds
Specifies a number in the range 0 through 59; the default value is 0.

hundredths
Specifies a number in the range 0 through 99; the default value is 0.

Remarks

MS OS/2 *MULTIUSER* keeps track of time in a 24-hour format and uses the time information to update directory listings whenever you create or change a file.

You may use either colons (:) or periods (.) to separate the hours, minutes, and seconds. To separate hundredths, you must use a period.

If you type TIME by itself, MS OS/2 *MULTIUSER* displays the current time and then prompts you for a new time. If you do not want to change the time, just press **ENTER**.

Security Class Restrictions

Administrator and Operator classes have permission to change the time. User and Guest classes can query the time. An error message will be displayed if a user who is not authorized to do so attempts to change the time.

Example

To set the system time to 1:36 P.M., type the following:

```
TIME 13:36
```

TRACE (Control System Trace)

Description

Turns the system trace on or off.

Syntax

TRACE [ON | OFF] [*eventcode* | TDF[,*eventcode* | TDF] [...]]
[/P:*all* | /P:*processID*[,...]] [/C] [/S | /R]

Parameters

ON
Starts system-event tracing.

OFF
Stops system-event tracing.

eventcode
Specifies a single event by using a decimal code number in the range 0 through 255. You use event codes to selectively turn on or off the tracing of specific system events. You can specify more than one event code, separating them with commas.

TDF
Specifies a trace definition file. The *filename* must not include a drive, path, or *filename* extension.

Options

/P:*all*
Turns tracing on or off for all present and future processes in the system.

/P:*processID*

Turns tracing on or off for only the processes specified. The system identifies a process by its hexadecimal process identification number (*processID*). You can specify more than one process by listing all their process identification numbers, separated by commas.

/C

Clears the current contents of the trace buffer.

/S

Suspends tracing temporarily but keeps the current specified set of trace points. This option cannot be used with the /R option.

/R

Resumes tracing with the currently specified trace points. This option cannot be used with the /S option.

Remarks

The system trace records actions, such as hardware interrupts or system functions, that MS OS/2 *MULTIUSER* has taken or processed while running. These actions are known as events and are identified by event codes.

The TRACE utility is similar to the trace configuration command. However, the trace utility cannot create a trace buffer in which to store the trace information. Therefore, the utility does not work unless either the TRACE or the TRACEBUF configuration command is in your *CONFIG.SYS* file when you start your system.

If you specify an invalid event code, MS OS/2 *MULTIUSER* still traces the other events listed but also displays an error message.

Security Class Restrictions

Only an Operator or an Administrator can execute this utility.

Example

To trace events 0 and 1, and no other system events, first type the following to turn off the system trace:

```
TRACE  OFF
```

Then type the following to turn on the trace for the events you want recorded:

```
TRACE ON 0,1
```

TRACEFMT (Display Trace Information)

Description

Displays the contents of the system-trace buffer.

Syntax

```
TRACEFMT
```

Remarks

The TRACEFMT utility analyzes and formats each trace record in the system-trace buffer and then displays the formatted trace records. (You may want to redirect the TRACEFMT output to a file.) The utility displays these formatted trace records, which consist of heading information and other trace-event information, in reverse chronological order, starting with the most recent event.

You must create the system-trace buffer when you start your system; to create the buffer, include either the TRACE or the TRACEBUF configuration command in your *CONFIG.SYS* file. If you have not done this, there is no buffer and the TRACEFMT utility has nothing to display.

TREE (Display Drive Directory Structure)

Description

Displays the path of each directory on a disk and lists all the subdirectories (and, as an option, all the files as well).

Syntax

TREE [*drive*:] [/F]

Parameters

drive:
Specifies the *drive* that contains the disk for which you want to display the directory structure. If you do not specify a *drive*, the utility displays the directory structure of the disk in the current drive.

/F
Lists the names of all the files in each directory and subdirectory on the disk.

Remarks

TREE displays only the paths and subdirectories for which you have Execute (X) or Read (R) permission.

Example

To display a list of all the directories and files on the disk in drive B, one screenful at a time, type the following:

```
TREE  B:  /F | MORE
```

TYPE (Display File)

Description

Displays the contents of one or more text files, in the order specified, without modifying them.

Syntax

TYPE [*drive*:][*path*]*filename* [...]

Parameters

filename
Specifies the name of the text file you want displayed. You can specify multiple filenames and you can use wildcard characters to specify groups of files with similar names.

Remarks

If you use the TYPE command to display a file that contains tabs, MS OS/2 *MULTIUSER* expands all the tabs to eight spaces.

If you use the TYPE command to display a binary file or a file created by an application, you may see unusual characters, in addition to the text.

You must have Read (R) permission to TYPE a file.

Example

To display the contents of the file *REPORT.JAN* from the disk in drive B, type the following:

```
TYPE B:REPORT.JAN
```

UNPACK (Decompress File)

Description

Decompresses and copies a compressed file.

Syntax

UNPACK [*drive*:][*path*]*filename* [*drive2*:][*path2*] [/V]

Parameters

filename
Specifies the name of the compressed file. This *filename* has an @ symbol as the third character of its extension.

drive2:
Specifies the drive you want the files to be copied to. If you do not specify a drive, the UNPACK utility uses the current drive.

path2

Specifies the directory you want the files to be copied to. If you do not specify a directory, the UNPACK utility uses the current directory on the specified drive. The utility always uses the original filename and extension as the destination *filename*.

Option

/V

Checks whether the sectors that the files were written to can be read.

Remarks

The MS OS/2 *MULTIUSER* installation program uses the COPY command to copy uncompressed files from the installation disks to your system and uses the UNPACK utility to decompress and copy compressed files — those with an @ symbol as the third character of their extension.

The UNPACK utility also copies files that are not compressed, so you can use UNPACK to copy a disk that contains both kinds of files.

The UNPACK utility uses the date, time, and any file attributes of the original compressed file in the header of the uncompressed file.

You must have Read (R) permission on the source file and Create/Modify (C/M) permission on the target file.

Example

To decompress the compressed files on a floppy disk in drive A that contains both compressed and uncompressed files and then copy all the files on the disk to the root directory of your fixed disk (drive C), type the following:

```
UNPACK A:\* C:\
```

VER (Display Version Number)

Description

Displays the MS OS/2 *MULTIUSER* version number.

Syntax

```
VER
```

VERIFY (Verify Disk/Diskette Operations)

Description

Turns write verification on or off.

Syntax

```
VERIFY [ON | OFF]
```

Remarks

When VERIFY is on, each time MS OS/2 *MULTIUSER* writes a file to a disk it checks whether the sectors the information was written to can be read (that no information was written to bad sectors, for example) and displays an error message if it cannot successfully write the file to the disk. (Verifying does not mean comparing the information that MS OS/2 *MULTIUSER* just wrote with the information that MS OS/2 *MULTIUSER* just read.) The default setting is off.

If you type VERIFY by itself, MS OS/2 *MULTIUSER* displays the current verification setting.

This command affects only the current session.

VOL (Volume Label)

Description

Displays the volume label and the volume serial number of the specified disk, if they exist.

Syntax

VOL [*drive*:] [...]

Parameter

drive:

Specifies the *drive* containing the disk for which you want to know the label and serial number. You can see the labels and serial numbers of more than one disk by specifying more than one drive. If you do not specify a *drive*, MS OS/2 *MULTIUSER* displays the volume label and serial number of the disk in the current drive.

Remarks

If you are displaying volume labels and serial numbers for multiple drives and MS OS/2 *MULTIUSER* cannot display this information for one of them (for example, if one of the drives you specified does not exist), MS OS/2 *MULTIUSER* displays an error message and continues displaying the information for the rest of the drives.

Disks formatted under versions of MS-DOS earlier than 4.0 do not have serial numbers.

You must have access to the fixed disk.

Example

To find out what the volume labels and serial numbers are for the disk in drive A and for your fixed disk (drive C), type the following:

```
VOL A: C:
```

XCOPY (Copy Groups of Files and Subdirectories)

Description

Copies files and directories, including any subdirectories, from one disk to another.

Syntax

XCOPY [*drive1*:][*path1*][*filename1*] [*drive2*:][*path2*][*filename2*] [/S] [/E] [/P] [/V] [/A] [/M] [/F] [/D:*date*]

Parameters

drive1:

Specifies the drive that contains the disk from which you want to copy files (the source disk). You can use this argument by itself, with *path1*, or with *path1* and *filename1*. If you specify only *drive1*, the XCOPY utility copies all the files in the current directory on that drive.

path1

Specifies the directory where the files you want to copy are located. If you specify *path1* without *drive1* or *filename1*, XCOPY copies all files in the specified directory on the current drive.

filename1

Specifies the name of the file you want to copy. You can use wildcard characters to copy multiple files with similar names.

drive2:

Specifies the drive that contains the disk to which you want to copy files (the destination disk). You can use this argument by itself, with *path2*, or with *path2* and *filename2*. If you specify only *drive2*, the XCOPY utility copies files to the current directory on that drive and uses the original *filenames*.

path2

Specifies the directory you want the files copied to. If you specify *path2* without *filename2*, XCOPY uses the original *filenames*.

filename2

Specifies the *filename* to which you want to copy the file. You can use wildcard characters to give multiple files similar names.

Options

/S

Copies directories and subdirectories unless they are empty. If you omit this option, XCOPY works within a single directory.

/E

Copies all subdirectories even if they are empty. If you use this option, you must also use the /S option.

/P

Prompts you to confirm that you want to create each new file.

/V

Checks whether the sectors that the files and directories were written to can be read.

/A

Copies files that have their archive bits set, without modifying the archive bit of the original file.

/M

Copies files that have their archive bits set and turns off the archive bit of the original file.

/F

Specifies that the XCOPY utility should not discard the extended attributes of a file if the destination file system does not support extended attributes. In this case, the utility does not copy the file.

/D:*date*

Copies files that were modified on or after the specified date. The format of date depends on the country code you are using; the default format is mm-dd-yy.

Remarks

All files or directories created by the XCOPY utility have the same extended attributes as the original files or directories.

If you omit the /F option when you use the XCOPY utility to copy a file with extended attributes to a file system that does not support extended attributes, the utility copies the file and discards the extended attributes. If the file requires the extended attributes, XCOPY does not copy the file but displays an explanatory message on the screen.

You must have Read (R) permission on the source and Create/Modify (C/M) permission on the target file.

Example

To copy all the files, directories, and subdirectories, even empty ones, on the disk in drive A to the disk in drive B and to verify each file as it is copied, type the following:

```
XCOPY A:\ B: /S /E /V
```

CHAPTER 3

CONFIGURATION COMMANDS *(CONFIG.SYS)*

AUTOFAIL (Popup Error Control)

Description

Turns on or off the mechanism by which MS OS/2 *MULTIUSER* displays a full screen error message that requires a choice of actions whenever the system encounters a hard error or an exception condition. To use this command, place it in your *CONFIG.SYS* file.

Syntax

AUTOFAIL=YES | NO

Parameters

YES
: Turns on the autofail mechanism, preventing MS OS/2 *MULTIUSER* from displaying a full-screen error message for a hard error or exception condition. MS OS/2 *MULTIUSER* returns an error code to the program instead. You may still see a command-line error message.

NO
: Turns off the autofail mechanism, allowing MS OS/2 *MULTIUSER* to display a full-screen error message for a hard error or exception condition. This is the default setting.

Remarks

A hard error is caused by something external to the software, such as an open disk-drive door or an incorrectly formatted floppy disk. An exception condition is caused by an instruction that the central processing unit (CPU) cannot execute, such as dividing by zero.

Resource Management also uses this mechanism to warn you that you have reached the end of your resources. Setting AUTOFAIL=YES disables these popups and an appropriate error return code is returned to the application.

When the AUTOFAIL mechanism is turned off, the system stops and displays a full-screen error message when a hard error or exception condition occurs; you must choose an action from a list in the error screen before the system will resume processing commands. When the AUTOFAIL mechanism is turned on, the system continues processing the command that caused the error.

BUFFERS (Disk Buffering)

Description

Sets the number of buffers in memory for use by the FAT file system. To use this command, place it in your *CONFIG.SYS* file.

Syntax

BUFFERS=*number*

Parameter

number

Specifies the number of buffers available. This must be a number in the range 1 through 100. During system installation, the value of the BUFFERS command is set to 30. If you remove the BUFFERS command from your *CONFIG.SYS* file, the system sets the value to 3.

Remarks

Buffers are work areas MS OS/2 *MULTIUSER* uses to hold data when it is reading from or writing to a disk. You can speed up your system's performance by increasing the number of buffers available, but when you do so you also reduce the amount of memory available.

Example

To create 20 disk buffers, include the following line in your *CONFIG.SYS* file:

```
BUFFERS=20
```

CACHE (HPFS Caching)

Description

See CACHE in the Commands section. CACHE is used in the *CONFIG.SYS* file for lazy writing.

Syntax

```
CACHE [/LAZY:ON | /LAZY:OFF]
```

Parameter

/LAZY:ON | /LAZY:OFF
Turns lazy writing on or off for all disks or partitions that are formatted for HPFS. The default setting is /LAZY:ON.

CALL (Run Program During System Initialization)

Description

Starts and runs a program in the foreground when you start your system. If you want to run a program that requires input from you, use this command rather than the RUN command. To use the CALL command, place it in your *CONFIG.SYS* file.

Syntax

CALL=[*drive*:][*path*]*filename* [*arguments*]

Parameters

filename
Specifies the program you want to start. You must include the extension (*.COM* or *.EXE*) and you must specify the *drive* and *path* if the file is not located in the root directory of your start-up drive. The file cannot be a batch file.

arguments
Specifies any valid options or other variables for the program you are calling.

Remarks

When you use the CALL configuration command to run a program, MS OS/2 *MULTIUSER* treats everything the program does as a single command; once that command has been carried out (that is, once the program has started, received any input needed from you, done whatever is necessary, and ended), MS OS/2 *MULTIUSER* resumes processing the *CONFIG.SYS* file where it left off.

You can include more than one CALL command in your *CONFIG.SYS* file.

MS OS/2 *MULTIUSER* processes all device commands in your *CONFIG.SYS* file before it starts processing CALL commands.

You can use the CALL configuration command to start monitor programs for other sessions but not for the session from which you are using CALL.

Example

To run the CHKDSK utility for drive D each time you start your system, when *CHKDSK.COM* is in the OS2 directory on drive C, include the following line in your *CONFIG.SYS* file:

```
CALL=C:\OS2\CHKDSK.COM  D:
```

CODEPAGE (Select Character Sets)

Description

Selects the code pages that the system will use. To use this command, place it in your *CONFIG.SYS* file.

Syntax

CODEPAGE=*xxx*[,*yyy*]

Parameters

xxx

Specifies the first code page. This must be a three-digit number from the list under "Remarks.".

yyy

Specifies the optional second code page. This must be a three-digit number from the list under "Remarks."

Remarks

If you specify two code pages, you can switch between them by using the CHCP command.

Code page, keyboard, and country are interrelated. A code page is a set of characters that are available to your system for use on the screen, for printing, and for sending to any other sort of output device. Your keyboard layout tells your system which characters from the character set correspond to which keystrokes; this can be different from country to country. The country you are working in (or for which you want to set up your system) determines which two code pages you should use. (For a list of country codes and their corresponding code pages, see the COUNTRY command.) You can change keyboard layouts without having to change code pages.

MS OS/2 *MULTIUSER* supports the following two code pages:

Code Page	Character Set
437	United States
850	Multilingual

The United States code page (437) includes most of the characters needed for most Western European languages but the multilingual code page (850), which is also used by systems other than personal computers, is more versatile and more complete. You will usually specify code page 850 as the second code page, no matter what national code page you specified first.

If you are setting up a keyboard, screen, or printer to use code pages, you must also use the DEVINFO command to specify the code pages the outside device is to use. For more information, see the DEVINFO configuration command.

Refer to the "Setting Up Code Page Support" section in Chapter 9 of the *Citrix MULTIUSER System Administrator's Guide.*

Example

To set up your system to use code pages 437 and 850, include the following line in your *CONFIG.SYS* file:

```
CODEPAGE=437,850
```

COUNTRY (Select Country)

Description

Specifies the country for which MS OS/2 *MULTIUSER* is to set up the system. To use this command, place it in your *CONFIG.SYS* file.

Syntax

COUNTRY=*xxx*[,[*drive*:][*path*]*filename*]

Parameters

xxx
Specifies the country by using a three-digit country code. You must include all three digits, even if the code begins with a zero. The default setting is 001 (United States).

filename

Specifies the file that contains information on country conventions and supported code pages. If you do not specify a drive or path, MS OS/2 *MULTIUSER* looks for this file in the root directory of the startup drive. If you do not specify *filename*, MS OS/2 *MULTIUSER* uses the default *COUNTRY.SYS* file in the root directory of the drive from which you started MS OS/2 *MULTIUSER*.

Remarks

The country you choose determines the MS OS/2 *MULTIUSER* conventions for such things as time and date format, decimal separators, and the order in which the SORT utility sorts ASCII characters. The country you specify with the country command also determines which code pages you should specify with the CODEPAGE command.

The country or language you can specify with the COUNTRY command, the corresponding country code, and the code pages supported for each country are:

Country/Language Group	Country Code	Code Pages
United States	001	437,850

The first of the two code pages for each country is the default code page for that country. If you do not include the CODEPAGE command in your *CONFIG.SYS* file, MS OS/2 *MULTIUSER* uses the system default code page.

Refer to the "Setting Up Code Page Support" section in Chapter 9 of the *Citrix MULTIUSER System Administrator's Guide*.

Example

To set up your system for the United States and instruct MS OS/2 *MULTIUSER* to look for *COUNTRY.SYS* in the OS2 directory on the disk in drive A, include the following line in your *CONFIG.SYS* file:

```
COUNTRY=001,A:\OS2\COUNTRY.SYS
```

DEVICE (Initialize Device Driver)

Description

Instructs MS OS/2 *MULTIUSER* to load the specified device driver. To use this command, place it in your *CONFIG.SYS* file.

Syntax

DEVICE=[*drive*:][*path*]*filename* [*arguments*]

Parameters

filename
Specifies the name of the file that contains the device driver. If this file is not in the root directory of the start-up drive, you must include the *drive* and/or *path*.

arguments
Specifies any valid options or other variables for the designated device driver.

Remarks

Each device connected to your system needs its own device driver. Each driver requires a separate DEVICE command in your *CONFIG.SYS* file.

You generally receive a device driver on a disk when you buy a new device; be sure that you place the device driver in the directory you specify with the DEVICE command.

MS OS/2 *MULTIUSER* processes device commands in the order in which they appear in your *CONFIG.SYS* file and before it processes any RUN commands in the file.

DEVINFO (Set Up Device for Codepage Switching)

Description

Prepares a device to use code pages. To use this command, place it in your *CONFIG.SYS* file.

Syntax

DEVINFO=
devtype,*subtype*,[*drive*:][*path*]*filename*[,ROM=[[(]*xxx*[,*yyy*)]][,...]]

Parameters

devtype
Specifies the type of device: keyboard, monitor, or parallel printer. See the list under "Remarks" for possible values.

subtype

Specifies the style or model of the device. For a keyboard, this argument would specify the keyboard layout. See the list under "Remarks" for possible values.

filename

Specifies the file that contains information about the code pages for that device. See the list under "Remarks" for possible values.

ROM=

Specifies that code pages are available to a printer, either in the printer's read-only memory or in a cartridge. This and the following options apply only to parallel printers.

xxx

Specifies a code page that is available for a parallel printer. Each code page is identified by a three-digit number; for a list of the possible code pages, see the CODEPAGE command. A printer may support more than one code page.

yyy

Specifies a font identification number that identifies a font on a parallel printer and associates that font with a particular code page. A code page may have more than one font associated with it. See your printer manual for font identification numbers.

Remarks

You must include a separate DEVINFO command in your *CONFIG.SYS* file for each device connected to your system, including the keyboard and the monitor, if you want to be able to switch code pages. The DEVINFO command specifies the kind of device you have connected to your system and the location of the code page or keyboard information for that device.

The MS OS/2 *MULTIUSER* installation program automatically places certain DEVINFO commands in your *CONFIG.SYS* file.

The following list shows the values you can give to *devtype*, *subtype*, and *filename*:

Argument	Keyboard	Monitor	Printer
DEVTYPE	KBD	SCR	PRN, LPT1, LPT2, LPT3
SUBTYPE	keyboard code	EGA, VGA	4201, 5202
FILENAME	KEYBOARD.DCP	VIOTBL.DCP	4201.DCP, 5202.DCP

Keyboard code is a two-letter code that identifies the keyboard layout for a particular country.

Examples

To prepare your keyboard to use the code pages you have specified with the CODEPAGE command, to use the United States keyboard layout, and to specify that the file containing code page information is in the OS2 directory on your fixed disk (drive C), include the following line in your *CONFIG.SYS* file:

```
DEVINFO=KBD,US,C:\OS2\KEYBOARD.DCP
```

To prepare an IBM Quietwriter III to use code pages 437 and 850, with multiple fonts, include the following line in your *CONFIG.SYS* file (type this as a single line, even though it appears here on more than one line):

```
DEVINFO=LPT1,5202,5202.DCP,ROM=(437,011),
(437,085),(437,254),(437,159),(850,254),
(850,159)
```

DISKCACHE (Allocate Cache Space)

Description

Enables disk caching for the FAT file system and specifies the amount of memory to be set aside for the disk cache. To use this command, place it in your *CONFIG.SYS* file.

Syntax

DISKCACHE=*n*[,*m*]

Parameters

n

Specifies the amount of memory, in kilobytes, to be set aside for the disk cache. This number must be in the range 64 through 7200.

m

Specifies the disk cache threshold value, in sectors. If the data exceeds *m* sectors, MS OS/2 *MULTIUSER* does not store it in the disk cache. This value must be in the range 1 through 32; the default value is 7.

Remarks

A disk cache is an extra buffer in which MS OS/2 *MULTIUSER* stores information that it has recently read from your fixed disk. When an application needs to read information from the fixed disk, it looks first in the disk cache to see if the information is there. Since it is much faster to read from the disk cache than to read from the fixed disk, disk caching can speed up your system. However, the disk cache uses part of system memory, so less memory is available to an application.

MS OS/2 *MULTIUSER* uses part of the memory set aside for the disk cache for control information. The amount of memory required for control information depends on the size of your fixed disk.

To change the size of the disk cache, change the DISKCACHE command in your *CONFIG.SYS* file and then restart your system.

You should use the DISKCACHE configuration command only if your system has a File Allocation Table (FAT) file system on a fixed disk.

Example

To set aside 128 kilobytes of memory for disk caching, include the following line in your *CONFIG.SYS* file:

```
DISKCACHE=128
```

HOSTNAME (Specify Name of System)

Description

Specifies the hostname which MS OS/2 *MULTIUSER* uses to identify the system.

Syntax

HOSTNAME=*hostname*

Parameter

hostname
Specifies the system name.

Remarks

The specified *hostname* is displayed by the QUERY HOST command.

The MS OS/2 *MULTIUSER* installation program automatically places a HOSTNAME command in your *CONFIG.SYS* file.

IFS (Install File System)

Description

Installs the driver for an installable file system and specifies the amount of memory to be reserved for disk caching. To use this command, place it in your *CONFIG.SYS* file.

Syntax

IFS=[*drive*:][*path*]*filename* [/C[ACHE]:*nnnn*]

Parameters

filename

Specifies the name of the file that contains the file system driver. If this file is not in the root directory of the start-up drive, you must include the drive and/or path.

/C[ACHE]:*nnnn*

(HPFS only) Specifies the amount of memory, in kilobytes, to be set aside for disk caching by the installable file system. You can abbreviate /cache as /c. The value of *nnnn* must be a number in the range 64 through 2048 and should be a multiple of 2, since HPFS divides the cache into 2-kilobyte blocks. If you specify an odd number when you are installing the HPFS driver, MS OS/2 *MULTIUSER* rounds it down to the next even number. The default cache size is either 64K or 10 percent of the total available memory, whichever is greater.

NOTE: This is not the CACHE utility which is for lazy writing.

Remarks

The IFS command should precede any device command in your *CONFIG.SYS* file except device commands that load device drivers required by the installable file system.

Example

To set aside 500 kilobytes of memory for HPFS disk caching, include the following line in your *CONFIG.SYS* file:

```
IFS=C:\OS2\HPFS.IFS /C:500
```

NOTE: The CACHE utility for HPFS lazy writing is used in the *CONFIG.SYS* file as follows:

RUN=C:\OS2\CACHE.EXE [*parameters*]

IOPL (Input/Output Privilege)

Description

Specifies whether MS OS/2 *MULTIUSER* can give data input/output privilege to a process that requests it in an MS OS/2 *MULTIUSER* session. To use this command, place it in your *CONFIG.SYS* file.

Syntax

IOPL=YES | NO | *program*[,...]

Parameters

YES
: Allows MS OS/2 *MULTIUSER* to give input/output privilege to a process.

NO
: Prevents MS OS/2 *MULTIUSER* from giving input/output privilege to a process. This is the default setting.

program
: Specifies the program that will be granted input/output privilege on request. You can specify more than one program, separating the names with commas.

Remarks

Some MS OS/2 *MULTIUSER* applications need direct access to hardware such as the display adapter. The IOPL command specifies whether they can receive that access. Avoid using IOPL=YES because the system security features are rigidly restrained. IOPL=YES allows any application with privilege level 2 to access input/output instructions. It is recommended that you specify *programs* on an individual basis. You should use the IOPL command only if an application requires you to do so.

LIBPATH (Location of Dynamic Link Libraries)

Description

Specifies the directories MS OS/2 *MULTIUSER* is to search for dynamic-link libraries. To use this command, place it in your *CONFIG.SYS* file.

Syntax

LIBPATH=[*drive*:]*path*[;[*drive*:]*path*][...]

Parameters

drive:
: Specifies the drive where dynamic-link libraries are located. If you do not specify a drive, MS OS/2 *MULTIUSER* searches the disk in the current drive.

path
: Specifies the directory to search for dynamic-link libraries. You can specify more than one directory, separating the names with semicolons (;).

Remarks

The MS OS/2 *MULTIUSER* installation program places the following line in your *CONFIG.SYS* file:

```
LIBPATH=C:\OS2\DLL;C:\OS2\DLL\CTX;C:\;
```

MS OS/2 *MULTIUSER* does not automatically search the current directory for dynamic-link libraries; you can specify the current directory by substituting a period (.) for the first drive:path argument.

Example

To instruct MS OS/2 *MULTIUSER* to look for dynamic-link libraries in the current directory and in the directory DYNLIB on your fixed disk (drive C), include the following line in your *CONFIG.SYS* file:

```
LIBPATH=.;C:\DYNLIB
```

LOG (Record System Events)

Description

Turns system-event logging on or off. When logging is on, MS OS/2 *MULTIUSER* records system events in the system log file. To use this command, place it in your *CONFIG.SYS* file.

Syntax

LOG=ON | OFF [/M:*minfree*] [/B:*buffersize*]

Parameters

ON
  Starts system-event logging.

OFF
  Stops system-event logging. This is the default setting.

Options

/M:*minfree*
Specifies the minimum amount of disk space, in kilobytes, to be left free (that is, not used for system-event logging) on your fixed disk. The default value is 512.

/B:*buffersize*
Specifies the size, in kilobytes, of the buffer to be used for logging system events. The maximum size is 64K; the default value is 4.

Remarks

System Install places the LOG=ON command into *CONFIG.SYS* to override the LOG=OFF default.

All data being logged is written to:

\OS2\SYSTEM\LOG0001.DAT

When the system is started, the existing LOG001.DAT is opened and new log records are appended. It can be closed using the EVENTS utility.

For more information about System Event Logging, see Chapter 14 in the *Citrix MULTIUSER System Administrator's Guide.*

MAINTENANCE (Maintenance Mode)

Description

Specifies whether MS OS/2 *MULTIUSER* is to start in maintenance mode. To use this command, place it in your *CONFIG.SYS* file.

Syntax

MAINTENANCE=[ON | OFF][,*terminalname*[,[*drive*:][*path*] *filename* [*arguments*]]]

Parameters

ON
Enables maintenance mode.

OFF
Enables the multiuser capabilities of the system.

,terminalname
Defines the terminal to be used for maintenance mode. If this parameter is not specified, the *terminalname* defaults to the console.

filename
Specifies the program you want to start. If this parameter is not specified, the program specified by COMSPEC is used. To use this parameter, *terminalname* must also be specified on the command. The drive and path parameters are not required if the *filename* is located in a directory specified in the PATH command.

arguments
Specifies any valid options or other variables for the program.

Remarks

The MS OS/2 *MULTIUSER* installation program places the MAINTENANCE=OFF command in your *CONFIG.SYS* file so that you can use the capabilities of the multiuser system.

Examples

example 1: OFF
Maintenance mode is set off so that normal multiuser activities may take place.

```
MAINTENANCE=OFF
```

example 2: ON
Maintenance mode is set on so that the system starts up in maintenance mode. The console is used here as the default maintenance terminal.

```
MAINTENANCE=ON
```

example 3: terminal option
Maintenance mode is set on so that the system starts up in maintenance mode. Terminal PLACE00 is used here as the maintenance terminal.

```
MAINTENANCE=ON,PLACE00
```

example 4: program option

Maintenance mode is set on so that the system starts up in maintenance mode. Terminal TERM9 is used here as the maintenance terminal, with the program *MAINTSH.EXE* started as the maintenance program.

```
MAINTENANCE=ON,TERM9,C:\ADMIN\MAINT
\MAINTSH.EXE
```

MAXWAIT (System Scheduling)

Description

Sets the maximum time a process must wait before MS OS/2 *MULTIUSER* increases its priority. To use this command, place it in your *CONFIG.SYS* file.

Syntax

MAXWAIT= *x*

Parameters

x

Specifies the number of seconds a process must wait before it is given a higher priority. This number must be in the range 1 through 255; the default value is 3.

Remarks

When an active process has waited *x* seconds without running, the MS OS/2 *MULTIUSER* scheduler increases the priority of the process for one execution cycle (time slice). For more information about time slices, see the TIMESLICE command.

The MAXWAIT command has no effect if the PRIORITY command is set to ABSOLUTE.

The default value is adequate for most systems. Do not change the default value unless told to do so.

Example

To instruct MS OS/2 *MULTIUSER* to give priority to processes after they have waited 2 seconds, include the following line in your *CONFIG.SYS* file:

```
MAXWAIT=2
```

MEMMAN (System Memory Management)

Description

Specifies whether MS OS/2 *MULTIUSER* can swap memory segments between memory and disk and whether it can temporarily move segments. To use this command, place it in your *CONFIG.SYS* file.

Syntax

```
MEMMAN=[SWAP | NOSWAP][,][MOVE | NOMOVE]
```

Parameters

SWAP
    Allows swapping of segments.

NOSWAP
    Prevents swapping of segments.

MOVE
Allows moving of segments.

NOMOVE
Prevents moving of segments.

Remarks

You can use a space instead of a comma to separate the arguments of a MEMMAN command.

The MS OS/2 *MULTIUSER* installation program places a MEMMAN command in your *CONFIG.SYS* file. If you start MS OS/2 *MULTIUSER* from a fixed disk, the default setting is MEMMAN=SWAP,MOVE; if you start from a floppy disk, the default setting is MEMMAN=NOSWAP,MOVE.

If you allow MS OS/2 *MULTIUSER* to SWAP, then it can MOVE segments too, but the reverse is not automatically true.

For more information about swapping and moving memory segments, see Chapter 9 in the *Citrix MULTIUSER System Administrator's Guide.*

The SWAPDOS and NOSWAPDOS parameters are not supported.

Example

To prevent MS OS/2 *MULTIUSER* from swapping or moving data segments while you are running a time-dependent application, include the following line in your *CONFIG.SYS* file:

```
MEMMAN=NOSWAP,NOMOVE
```

PAUSEONERROR (Pause During System Initialization Errors)

Description

Specifies whether MS OS/2 *MULTIUSER* is to pause if it encounters an error while processing your *CONFIG.SYS* file during system start-up. To use this command, place it in your *CONFIG.SYS* file.

Syntax

```
PAUSEONERROR=YES|NO
```

Remarks

If PAUSEONERROR is set to YES (the default setting), then whenever MS OS/2 *MULTIUSER* encounters an error while processing your *CONFIG.SYS* file, it stops, displays an error message, and prompts you to press **ENTER** to continue the start-up process. If you have set PAUSEONERROR to NO, MS OS/2 *MULTIUSER* displays an error message when it encounters an error, but it does not stop.

Example

To instruct the system to process your *CONFIG.SYS* file without pausing when it encounters errors, include the following line in your *CONFIG.SYS* file:

```
PAUSEONERROR=NO
```

PRIORITY (System Scheduling)

Description

Specifies how a process receives enough priority over other processes to run. To use this command, place it in your *CONFIG.SYS* file.

Syntax

```
PRIORITY=ABSOLUTE | DYNAMIC
```

Parameters

ABSOLUTE

Prevents the system from dynamically changing the priority of processes in the general-priority category. The absolute setting allocates central-processing-unit (CPU) time according to the process's current priority.

DYNAMIC

Instructs MS OS/2 *MULTIUSER* to try to determine which process needs CPU resources most in any given interval of time (time slice). The dynamic setting gives more CPU time to the process that is running in the foreground. This is the default setting. For more information about time slices, see the TIMESLICE command.

Remarks

In MS OS/2 *MULTIUSER*, processes (threads) are divided into three categories according to their priority: general, time-critical, and low. The general-priority category is further divided into foreground, background, and interactive subcategories. Normally, MS OS/2 *MULTIUSER* automatically adjusts the priority levels of general-priority processes as circumstances change (for example, when a foreground process becomes a background process).

The default value is adequate for most systems. Do not change the default value unless told to do so.

PROTECTONLY (Protect Mode)

Description

System install places this configuration command in the *CONFIG.SYS* file.

Syntax

```
PROTECTONLY=YES
```

Remarks

PROTECTONLY=NO is not supported in MS OS/2 *MULTIUSER*.

REBOOT (System Console Hotkeys)

Description

Specifies whether MS OS/2 *MULTIUSER* is to enable the system restart and system dump hotkeys at the console. To use this command, place it in your *CONFIG.SYS* file.

Syntax

```
REBOOT=[ON | OFF]
```

Remarks

The MS OS/2 *MULTIUSER* installation program places the REBOOT=OFF command in your *CONFIG.SYS* file so that you can use the:

- **CTRL+ALT+DEL** hotkey at the console, to terminate all applications running on the console terminal and restart the console terminal. The other applications running in the system are not disturbed. To terminate all applications, use the SHUTDOWN command at the console.

- **CTRL+ALT+NUMLOCK** twice hotkey at the console is not enabled. No system dump is available.

To change the effect of these hotkeys on the console, change this line to REBOOT=ON then you can use the:

- **CTRL+ALT+DEL** hotkey at the console to terminate all applications running on the system and restarts the system. It is a good practice to always notify all system users before terminating all the applications using the console Restart Hotkey or the SHUTDOWN command.

- **CTRL+ALT+NUMLOCK** twice hotkey at the console to provide a system dump. This copies the contents of memory on a disk prepared using the CREATEDD command.

The REBOOT=OFF command option is used when the console is utilized as a terminal.

REM (Remark)

Description

Includes a remark or descriptive comment in your *CONFIG.SYS* file. Lines that begin with REM are not processed by MS OS/2 *MULTIUSER*.

Syntax

REM [*text*]

Parameter

text

Specifies the *remark* (or comment) that you want to include in your *CONFIG.SYS* file. Text may be any string of characters that fits on one line.

Remarks

If the comment you want to put in the *CONFIG.SYS* file is too long to fit on one line, you must use the REM command again for each line of the comment. You can also use REM without text to add spacing between blocks of comments or remarks.

Example

To leave a comment in a configuration file for another person's use or to remind yourself of the purposes of the commands in the file, begin each line of the comment with the REM command, as follows:

```
REM MS OS/2 MULTIUSER uses CONFIG.SYS to configure your
REM system. This command (REM) allows you to insert
REM comments in CONFIG.SYS that remind you what a
REM particular command is supposed to do.
```

RESOURCE (Resource Management Control)

Description

Specifies if resource management should enforce user limits. To use this command place it in the *CONFIG.SYS* file.

Syntax

```
RESOURCE=ON | OFF
```

Parameters

ON

If the RESOURCE command it is set to ON, resource management will check your limits to ensure a distribution of resources as defined by the System Administrator.

OFF

If the RESOURCE command is set to OFF, resource management will not check your limits.

Remarks

Turning resource management off allows all users to compete for resources such as memory, threads, semaphores, and file handles on a first-come first-serve basis.

System installation places the command RESOURCE=ON in *CONFIG.SYS*.

Example

To disable resource management of user limits add the following line into *CONFIG.SYS*:

```
RESOURCE=OFF
```

RUN (Run Background Program)

Description

Starts a background program when you start the system (if used in the *CONFIG.SYS* file) or when you log into the system (if used in the user's *CONFIG.USR* file).

Syntax

RUN=[(*loginname*)] [*drive*:][*path*]*filename* [*arguments*]

Parameters

(*loginname*)
Specifies the *loginname* under which the program will execute. When using this option, both the *username* and *groupname* must be specified for *loginname*. If the *loginname* parameter is omitted, the program will execute under the loginname specified for RUN commands in CONFIG SYSTEM (the system profile). This parameter is not supported in *CONFIG.USR*.

filename

Specifies the program you want to start, which must be one that can run in the background. The file cannot be a batch file. The drive and path parameters are not required if the *filename* is located in a directory specified in the PATH command.

arguments

Specifies any valid options or other variables for the program.

Remarks

RUN commands within the *CONFIG.SYS* file are executed when the system starts. RUN commands within the *CONFIG.USR* file are executed when the corresponding user logs into the system.

You can include more than one RUN command in your *CONFIG.SYS* or *CONFIG.USR* files.

When processing the RUN command, MS OS/2 *MULTIUSER* preserves the difference between uppercase and lowercase letters in arguments. This is important for some programs that are case sensitive.

Examples

example 1: *loginname*

This RUN command in *CONFIG.SYS* starts the program *STATS.EXE* in a login for the *loginname* SERVER.ADMIN.

```
RUN=(SERVER.ADMIN)  C:\ADMIN\APPS\STATS.EXE
```

example 2: no *loginname*

If located in *CONFIG.SYS*, this RUN command starts the program *STATS.EXE* in a login for the default *loginname* as specified by CONFIG SYSTEM (the system profile). If located in *CONFIG.USR*, this RUN command starts the program in the login for the current *loginname.*

```
RUN=C:\ADMIN\APPS\STATS.EXE
```

SECURITY (System Security)

Description

Specifies whether or not any security access checks should be made during the normal operation of the system. This command is only valid in *CONFIG.SYS.*

Syntax

```
SECURITY=ON | OFF
```

Parameters

ON

Specifies that the security system should be enabled and any established access restrictions should be enforced.

OFF

If OFF is specified, no access operation will be denied regardless of how security attributes are configured.

Remarks

The default is SECURITY=ON.

SECURITY=OFF should be used with great care. Much of the integrity of the system relies on the security system blocking users and programs from accessing certain files and system functions. For example, the system profiles are secured so that the system configuration cannot be inadvertently destroyed, requiring a re-install process. Additionally, programs and users are blocked from using certain system functions which could impact sessions of other users.

SET (Set Environment)

Description

Defines an environment variable by naming the variable and giving a value for it.

Syntax

SET *string1=string2*

Parameters

string1
Specifies the name of the environment variable you want to set (for example, PATH, INIT, LIB, or PROMPT).

string2
Specifies the string that defines the current value of the environment variable.

Remarks

The MS OS/2 *MULTIUSER* installation program places the following SET commands in your *CONFIG.SYS* file to specify the search paths to be used by *CMD* and other programs (PATH and DPATH) and the location of the command interpreter (COMSPEC):

```
SET PATH=C:\OS2;C:\OS2\CTX;C:\OS2\SYSTEM;
    C:\OS2\INSTALL;C:\;
SET DPATH=C:\OS2;C:\OS2\CTX;C:\OS2\SYSTEM;
    C:\OS2\INSTALL;C:\;
SET COMSPEC=C:\OS2\CMD.EXE
```

The PATH environment variable sets the path that MS OS/2 *MULTIUSER* uses to search for a command file or program file (files with extensions *.EXE*, *.COM*, or *.CMD*) if that file is not in the current directory.

The DPATH environment variable sets the path that MS OS/2 *MULTIUSER* uses to search for a data file (files with extensions other that *.EXE*, *.COM*, or *.CMD*) if that file is not in the current directory.

The COMSPEC environment variable identifies the command interpreter. The system uses COMSPEC to identify the program started for the Session Create hotkey, **ALT+TAB**. An application program may use this variable to identify the command interpreter to invoke when processing batch files or other commands.

For more information about environment variables, see Chapter 3 of the *Citrix MULTIUSER User's Guide.*

SWAPPATH (Swap File Information)

Description

Specifies the location of the SWAP file that temporarily holds the information being swapped from memory if you have enabled swapping between memory and disk. To use this command, place it in your *CONFIG.SYS* file.

Syntax

SWAPPATH=*drive*:[*path*] [*space*]

Parameters

drive:
Specifies the drive on which the SWAP file is located.

space
Specifies the minimum amount of disk space, in kilobytes, to leave free on the swap drive (that is, the disk space that may not be used for swapping). This number must be in the range 0 through 32767; the default value is 512.

Remarks

The SWAPPATH command takes effect only if the MEMMAN command in your *CONFIG.SYS* file is set to allow swapping.

For information about swapping, see the MEMMAN command.

Example

To instruct MS OS/2 *MULTIUSER* to put the swap file in the TEMP directory on your fixed disk (drive C) and to reserve 1024K of free disk space on the swap drive, include the following line in your *CONFIG.SYS* file:

```
SWAPPATH=C:\TEMP 1024
```

THREADS (Maximum Number of Threads)

Description

Specifies the number of threads MS OS/2 *MULTIUSER* can run at one time. To use this command, place it in your *CONFIG.SYS* file.

Syntax

THREADS=*x*

Parameter

x

Specifies the number of threads. This number must be in the range 64 through 1024; the MS OS/2 *MULTIUSER* installation program sets the number to 512. Because threads take up memory in your system, it is recommended that you do not increase the number of threads unless your application tells you to do so or you receive a message telling you that you have run out of threads.

Remarks

A thread is part of an application or other process that MS OS/2 *MULTIUSER* can schedule to run on its own. A process generally contains multiple threads, which act like small programs that perform particular tasks in each process. Approximately forty of the threads you specify with the THREADS command are system threads that MS OS/2 *MULTIUSER* uses (the number may vary, depending on how your system is configured). These threads are not available to applications.

TIMESLICE (System Scheduling)

Description

Sets the amount of time that MS OS/2 *MULTIUSER* allocates to a thread before checking the priority of other threads. Time slices are the units of time that MS OS/2 *MULTIUSER* uses to schedule its activities. To use this command, place it in your *CONFIG.SYS* file.

Syntax

TIMESLICE=*x*[,*y*]

Parameters

x

Specifies the minimum length of the time slice, in milliseconds; the default value is 32.

y

Specifies the maximum length of the time slice, in milliseconds. This number must be equal to or greater than *x*. If you do not specify *y*, MS OS/2 *MULTIUSER* uses *x* as the maximum length also. If you do not include the TIMESLICE command in your CONFIG.SYS file, the default value of *y* is 248.

Remarks

When TIMESLICE is set to 32 (the default value), the process that is running in the foreground receives the most central-processing-unit (CPU) time. For more information about this topic, see the MAXWAIT and PRIORITY commands and Chapter 9 of the *Citrix MULTIUSER System Administrator's Guide.*

The default value is adequate for most systems. Do not change the default value unless told to do so.

Example

To set the minimum length of the time slice to 45 milliseconds and the maximum length to 148 milliseconds, include the following line in your *CONFIG.SYS* file:

```
TIMESLICE=45,148
```

TRACE (System Trace)

Description

Turns the system trace on or off. To use this command, place it in your *CONFIG.SYS* file.

Syntax

TRACE=ON | OFF [*eventcode*[,*eventcode*] [...]]

Parameters

ON
Starts system-event tracing.

OFF
Stops system-event tracing. This is the default setting.

eventcode
Specifies a single event by using a decimal code number in the range 0 through 255. You use event codes to selectively turn on or off the tracing of specific system events. You can specify more than one event code, separating them with commas.

Remarks

The system trace records actions, such as hardware interrupts or system functions, that MS OS/2 *MULTIUSER* has taken or processed while running. These actions are known as events and are identified by event codes. The system-trace information can be useful if you are writing a program to run with MS OS/2 *MULTIUSER*.

If you do not specify an event code, the TRACE command turns the system trace on or off for all events.

If you specify an invalid event code, MS OS/2 *MULTIUSER* still traces the other events listed but also displays an error message.

If you include the trace command in your *CONFIG.SYS* file but do not use the TRACEBUF command to specify a size for the trace buffer, MS OS/2 *MULTIUSER* sets aside 4 kilobytes of memory (the default size) for the trace buffer.

You must include either the TRACE command or the TRACEBUF command in your *CONFIG.SYS* file in order for system tracing to work.

Example

To turn on the system trace for all events except those that have the event codes 31 through 34, include the following lines in your *CONFIG.SYS* file:

```
TRACE=ON
TRACE=OFF 31,32,33,34
```

TRACEBUF (Size of System Trace Buffer)

Description

Sets the size of the system-trace buffer, where MS OS/2 *MULTIUSER* stores information about system events that are being traced. To use this command, place it in your *CONFIG.SYS* file.

Syntax

TRACEBUF=*x*

Parameter

x

Specifies the size, in kilobytes, of the trace buffer. This number must be in the range 1 through 63; the default value is 4.

Remarks

If you include the TRACE command in your *CONFIG.SYS* file but do not use the TRACEBUF command to specify a size for the trace buffer, MS OS/2 *MULTIUSER* sets aside 4 kilobytes of memory (the default size) for the trace buffer.

You must include either the TRACE command or the TRACEBUF command in your *CONFIG.SYS* file in order for system tracing to work.

CHAPTER 4

DEVICE DRIVERS

COM0x.SYS (Serial Communications Device)

Description

Directs MS OS/2 *MULTIUSER* to load the *COM0x.SYS* device driver, which allows you to use your system's serial communications port. To use this command, place it in your *CONFIG.SYS* file.

Syntax

DEVICE=[*drive*:][*path*]COM0*x*.SYS

Parameter

x

Specifies the communications-port device driver to load. For an Industry Standard Architecture 386 compatible computer, *x* must be 1. For a Micro Channel Architecture or compatible computer, *x* must be 2.

Examples

To direct MS OS/2 *MULTIUSER* to load the communications-port driver for your Industry Standard Architecture 386 compatible computer, add the following line to your *CONFIG.SYS* file:

```
DEVICE=C:\OS2\COM01.SYS
```

To direct MS OS/2 *MULTIUSER* to load the communications-port driver for your Micro Channel 386 compatible computer, add the following line to your *CONFIG.SYS* file:

```
DEVICE=C:\OS2\COM02.SYS
```

EXTDSKDD.SYS (Diskette Drive)

Description

Directs MS OS/2 *MULTIUSER* to load the *EXTDSKDD.SYS* device driver. To use this command, place it in your *CONFIG.SYS* file.

Syntax

DEVICE=[*drive*:][*path*]EXTDSKDD.SYS [/D:*drive*] [/T:*tracks*] [/S:*sectors*] [/H:*heads*] [/F:*type*]

Options

/D:*drive*
Specifies the physical *drive* number. The number must be in the range 0 through 255. The first physical floppy-disk drive (drive A) is drive 0; a second physical floppy-disk drive is drive 1; a third physical floppy-disk drive, which must be external, is drive 2.

/T:*tracks*
Specifies the number of *tracks* per side of a block device. The number must be in the range 1 through 999; the default value is 80.

/S:*sectors*
Specifies the number of *sectors* per track. The number must be in the range 1 through 99; the default value is 9.

/H:*heads*
Specifies the number of disk read/write *heads*. The number must be in the range 1 through 99; the default value is 2.

/F:*type*

Specifies the *type* of drive. This value must be 0 (160/180K or 320/360K), 1 (1.2 megabyte), or 2 (720K), the default type.

Remarks

When the *EXTDSKDD.SYS* device driver is loaded, you can access a floppy disk by using a logical drive letter. You can associate the letter with an external disk drive or you can associate a second name (an alias) with an internal or external disk drive and copy to and from that same disk drive.

Examples

To associate an alias with an internal 1.2-megabyte drive A, include the following line in your *CONFIG.SYS* file:

```
DEVICE=C:\OS2\EXTDSKDD.SYS /D:0 /T:80 /S:15 /H:2 /F:1
```

To copy from the external disk drive to that same external drive, include the following lines in your *CONFIG.SYS* file:

```
DEVICE=C:\OS2\EXTDSKDD.SYS /D:2
DEVICE=C:\OS2\EXTDSKDD.SYS /D:2
```

The first line associates the next available drive letter with the external disk drive. The second line associates an additional drive letter (an alias) with that same external drive.

VDISK.SYS (Virtual Disk)

Description

Directs MS OS/2 *MULTIUSER* to load the *VDISK.SYS* device driver. To use this command, place it in your *CONFIG.SYS* file.

Syntax

DEVICE=[*drive*:][VDISK.SYS [*vdisk-size*]
[,*sector-size*][,*entries*]

Options

vdisk-size
Specifies the size, in kilobytes, of the virtual disk drive. The default size is 64K.

sector-size
Specifies the sector size, in bytes. This number must be 128, 256, or 512; the default value is 128.

entries
Specifies the number of directory entries. This number must be in the range 2 through 1024; the default value is 64.

Remarks

When the *VDISK.SYS* device driver is loaded, it creates a virtual disk drive — that is, a disk drive simulated in memory. Information stored in memory can be accessed quickly.

The maximum size of a virtual disk depends on the amount of available memory in your system, up to 4 megabytes. If the virtual-disk size specified is too large to fit in memory, the *VDISK.SYS* device driver will try to make a 16K virtual disk. This may result in a virtual disk with a different number of directory entries than you specified.

When you specify a value for the number of directory entries, MS OS/2 *MULTIUSER* rounds the value up to the nearest sector-size boundary. For example, if you give a value of 43 and your sector size is 512 bytes, MS OS/2 *MULTIUSER* rounds 43 up to 48, which is the next multiple of 16 (there are sixteen 32-byte directory entries in 512 bytes).

MS OS/2 *MULTIUSER* recognizes the *vdisk-size*, *sector-size*, and *entries* arguments by their positions. If you omit *vdisk-size* or *sector-size*, you must type a comma before the next argument as a placeholder. (If you include all three arguments, you can use spaces instead of commas.)

You must place the *DEVICE=VDISK.SYS* command after any *DEVICE=EXTDSKDD.SYS* commands in your *CONFIG.SYS* file.

Example

To set up a 64K virtual disk with 512-byte sectors and 32 directory entries, when *VDISK.SYS* is in the \OS12\DEV directory on drive C, include the following line in your *CONFIG.SYS* file:

```
DEVICE=C:\OS12\DEV\VDISK.SYS  ,512,32
```

APPENDIX A

COMMAND LINE INVOCATION OF THE CONFIG UTILITY

CONFIG ACCESS

Description

This utility configures system resources with security attributes, consisting of Access Control Lists (ACLs) and Audit Masks.

Syntax

CONFIG ACCESS [*resourcename*] [*accessname*] [*permissions*] [*options*]

Parameters

resourcename
Name of resource to query or modify. This can be:

- File or directory path
  Need not be fully qualified
- Device name
  Must end with : (such as LPT1:)

- Drive letter
  Must end with : (such as G:)

- Terminal name
  Must have \TERM\ as prefix
  (for example, \TERM\TERM01)

- User function point
  Must have \UFP\ as prefix
  (for example, \UFP\DO_THIS_COMMAND)

- Program function point
  Must have \PFP\ as prefix
  (for example, \PFP\IOCTL_CAT9)

- Application program interface
  Must have \API\ as prefix
  (for example, \API\DOSCALL.123)

accessname

Name being given access to resource or being deleted from access list of resource. This is not required for query and changes in audit flags. This can be:

- *Loginname*
  Must be in the format *username*[*.groupname*]. The *username* or *groupname* (or both) can be *, indicating that the permissions apply to all *usernames* or *groupnames*. If *groupname* is not specified, then the default group is used.

- Security class
  Can be Guest, User, Operator, or Administrator.

- Program *aliasname*
  This is the alias of a registered program or registered program path.

permissions

The list of permissions (letters) being given to the specified *accessname* for the specified *resourcename*. The list of valid letters is described below. If permissions already exist for the given *accessname*, this list will replace the current list. Permissions are not specified if deleting the entire entry.

Valid permission letters are:

- All resources

*	All applicable permissions
N	none (allow no access)

- Files and Directories

R	Read
W	Write
C	Create
D	Delete
X	Execute
A	Change attributes

- Drives and Character Devices

U	Use
V	Reserve

- Terminals

L	Login

- User Function Points

S	Query Self
O	Query Other
M	Modify Self
T	Modify Other
X	Execute

- Program Function Points and APIs

X	Execute

Options

/G+:*accessletters*

Specifies to add this list of access types (*accessletters*) to those being audited when access is granted. The *accessletter* is equal to the *permissions* letters.

/G-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is granted. The *accessletter* is equal to the *permissions* letters.

/D+:*accessletters*

Specifies to add this list of access types (*accessletters*) to those being audited when access is denied. The *accessletter* is equal to the *permissions* letters.

/D-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is denied. The *accessletter* is equal to the *permissions* letters.

/D

Deletes the ACL entry specified by *accessname.*

/Q

Queries the ACL and audit information of the specified *resourcename.*

/DACL

Deletes the ACL associated with the *resourcename.*

/F

Used with the /DACL option to force the ACL to be deleted without user confirmation.

/DAUDIT

Deletes the AUDIT flags associated with the *resourcename.*

/? (help)

Displays the syntax for the utility and information about the utility's options.

Security Class Restrictions

Administrators can view and change all security attributes. Operators can view all security attributes but can only change permissions for the files they own. Users and Guests can view security attributes only for those resources to which they have access, and can change only those that they own.

CONFIG GROUP

Description

These are the command line *functions* and *options* for the CONFIG GROUP utility. The CONFIG GROUP utility is typically run by going to the full screen (no *functions* or *options* specified). If *options* are specified with no *functions*, group profile *groupname* will be modified.

Syntax

CONFIG GROUP *groupname* [*functions*] [*options*]

Functions

/DEL

Deletes the group profile named *groupname.* This *function* requires user confirmation before the profile is deleted unless the /F option is used.

/NEW

Creates a new group profile named *groupname*. Any of the options defined below can be used to override the defaults.

/Q

Queries the *groupname* group profile and directs the output to STDOUT.

/REN:*NewGroupname*

Renames the *groupname* group profile to *NewGroupname.*

Options

/F

Used with the /DEL function to force the group profile delete without requiring user confirmation.

/G+:*accessletters*

Specifies to add this list of access types (*accessletters*) to those being audited when access is granted. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/G-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is granted. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/D+:*accessletters*

Specifies to add this list of access types (*accessletters*) to those being audited when access is denied. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/D-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is denied. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/WORKDIR:*directory*

Redefines the working directory for member users to be *directory*. The working directory is used to override the member user's home directory as the default current directory when a member user logs in to the system.

/USERPROGRAM:*programName*

Redefines the first user program to *programName*. The first user program is the program that gets executed first when a member user successfully logs in to the system.

/CLASSLEVEL:*levelname*

Redefines the class level to *levelname* for member users.

/MAXLOGINS:*number*

Redefines the default maximum times each member user can be logged in at any given time to *number*.

/MAXSESSIONS:*number*

Redefines the default maximum sessions allowed for each member user in any login to *number*.

/MAXFILES:*number*

Redefines the default maximum file handles allowed for each member user to *number*.

/MINFILES:*number*

Redefines the default minimum file handles that must be available before member users can login to *number*.

/MAXMEMORY:*number*

Redefines the default maximum amount of virtual memory (in Kb) allowed for each member user to *number.*

/MINMEMORY:*number*

Redefines the default minimum amount of virtual memory (in Kb) that must be available before member users can login to *number.*

/MAXSEMAPHORES:*number*

Redefines the default maximum number of system semaphores that each member user can have to *number.*

/MINSEMAPHORES:*number*

Redefines the default minimum number of system semaphores that must be available before member users can login to *number.*

/MAXTHREADS:*number*

Redefines the default maximum number of threads that each member user can have to *number.*

/MINTHREADS:*number*

Redefines the default minimum number of threads that must be available before member users can login to *number.*

/LOGIN*day:starttime,endtime*

Changes the login restriction of the day of the week to be from *starttime* to *endtime*. The login restriction is used to restrict when a user can be logged into the system. Valid times range from 00:00 to 24:00. Valid *days* are (SUN, MON,TUE,WED,THU,FRI,SAT). If the *starttime* is greater than the *endtime*, the time span will cross midnight

(24:00) and carry over into the next day. For example, if the option "/LOGINMON:17:00,6:00" is used, the user can login to the system from 5:00pm on Monday through 6:00am on Tuesday.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Security Class Restrictions

Administrator class has full access to all the CONFIG *functions.* Operator class can view all configurations but cannot modify any. User and Guest classes can view only their User profiles and related Group defaults; they cannot modify anything.

CONFIG SYSTEM

Description

These are the command line *functions* and *options* for the CONFIG SYSTEM utility. The CONFIG SYSTEM utility is typically run by going to the full screen (no *functions* or *options* specified). If *options* are specified with no *functions*, the system profile will be modified.

Syntax

CONFIG SYSTEM [*functions*] [*options*]

Functions

/NEW
Creates the system profile.

/Q

Queries the system profile and directs the output to STDOUT.

Options

/G+:*accessletters*

Specifies to add this list of access types (*accessletters*) to those being audited when access is granted. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/G-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is granted. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/D+:*accessletters*

Specifies to add this list of access types (*accessletters*) to those being audited when access is denied. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/D-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is denied. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/EVENTS+:*eventletters*

Specifies to turn on logging of the events specified by *eventletters*. The *eventletters* are described in CHANGE EVENTS.

/EVENTS-:*eventletters*

Specifies to turn off logging of the events specified by *eventletters*. The *eventletters* are described in CHANGE EVENTS.

/PWLIFE:*number*

Changes the life of a new password to *number* days. The life of a password is the number of days a password can be used until it must be changed. A zero (0) in this field indicates that passwords do not expire.

/PWMAX:*number*

Redefines the maximum length a *password* can be to *number*. Passwords cannot be more than 15 characters in length.

/PWMIN:*number*

Redefines the minimum length a *password* can be to *number*. Setting this to zero (0) indicates that users are not required to have *passwords*.

/PWWARN:*number*

Redefines the number of days before a *password* expires to start warning users to *number*. The warning is displayed after the login name is entered at the Login: prompt.

/PWSYNTAX:*string*

Redefines the *password* syntax definition to *string*. Each character in the string defines the character type that can be used in a *password*. The character positions in *string* correspond to the character positions in a *password*.

The following characters can be used to define the *password* syntax:

A	Alpha character
V	Vowel
C	Consonant
N	Numeric character
R	Alphanumeric character
X	Any character

The default is alphanumeric (R) for character positions not defined.

/HKLOGIN:*hotkeyString*
Redefines the default hotkey used to switch between user logins on the same terminal to *hotkeyString*.

/HKSESS:*hotkeyString*
Redefines the default hotkey used to switch between sessions within the same user login to *hotkeyString*.

/HKDIRECT:*hotkeyString*
Redefines the default hotkey used to go directly to the session that registers for the direct hotkey to *hotkeyString*. The program that registers for the direct hotkey is typically the Program Selector (*PSEL.EXE*).

/HKCREATE:*hotkeyString*
Redefines the default hotkey used to create a new session within the current login to *hotkeyString*.

/HKPRINT:*hotkeyString*
Redefines the default hotkey used to do a print screen to *hotkeyString*.

/HKPRINTTOGGLE:*hotkeyString*

Redefines the default hotkey used to toggle print screen on and off to *hotkeyString*.

/HKRESET:*hotkeyString*

Redefines the default hotkey used to reset the terminal to *hotkeyString*.

/HKCTRLBRK:*hotkeyString*

Redefines the default hotkey used as the **CTRL+BREAK** key sequence to *hotkeyString*.

/HKCTRLC:*hotkeyString*

Redefines the default hotkey used as the **CTRL+C** key sequence to *hotkeyString*.

NOTE: Refer to the "Hotkey Strings" section in CONFIG TERMINAL in the Appendix for information on how to write *hotkeyStrings.*

/USERPROGRAM:*programname*

Redefines the default first user program to *programname.* The first user program is the program that gets executed first when a user successfully logs in the system.

The first user program must contain the *.EXE* file extension. Typically, the Program Selector (*PSEL.EXE*) is the first user program.

Arguments can be specified with the *programname.* If *arguments* are used, surround the *programname* and *arguments* with double quotes (").

/AUTOLOGIN:*loginname*

Redefines the default autologin user that is used when a new terminal is created to *loginname.* A *password* can also be specified the same way it would be on the command line for the PASSWORD command

(PASSWORD:*password*). If a *password* is specified, surround the *loginname* and *password* with double quotes ("). If no autologin parameters are specified for a given terminal configuration, the terminal starts with the Login: prompt.

/RUNLOGIN:*loginname*

Redefines the default *loginname* for the system startup commands to *loginname.* Refer to Chapter 4, section "Using Startup Files" of the *Citrix MULTIUSER System Administrator's Guide.*

/RUNTERM:*terminalname*

Redefines the default *terminalname* for the system startup commands to *terminalname.* Refer to Chapter 4, section "Using Startup Files" of the *Citrix MULTIUSER System Administrator's Guide.*

/TIMEOUT:*number*

Redefines the number of minutes that a login must be inactive before it is automatically disconnected to *number.* A login is inactive when all sessions in the login are waiting on input from the keyboard.

/MAXLOGINS:*number*

Redefines the default maximum logins allowed for users to *number.*

/MAXSESSIONS:*number*

Redefines the default maximum sessions allowed for users in any login to *number.*

/MAXFILES:*number*

Redefines the default maximum file handles allowed for users to *number.*

/MINFILES:*number*
Redefines the default minimum file handles that must be available before a user can login to *number*.

/MAXMEMORY:*number*
Redefines the default maximum amount of virtual memory in kilobytes (Kb) allowed for users to *number*.

/MINMEMORY:*number*
Redefines the default minimum amount of virtual memory (in Kb) that must be available before a user can login to *number*.

/MAXSEMAPHORES:*number*
Redefines the default maximum number of system semaphores that users can have to *number*.

/MINSEMAPHORES:*number*
Redefines the default minimum number of system semaphores that must be available before a user can login to *number*.

/MAXTHREADS:*number*
Redefines the default maximum number of threads that users can have to *number*.

/MINTHREADS:*number*
Redefines the default minimum number of threads that must be available before a user can login to *number*.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Security Class Restrictions

Administrator class has full access to all the CONFIG *functions.* Operator class can view all configurations but cannot modify any. User and Guest classes do not have access to the system profile.

CONFIG TERMINAL

Description

These are the command line *functions* and *options* for the CONFIG TERMINAL utility. The CONFIG TERMINAL utility is typically run by going to the full screen (no *functions* or *options* specified). If *options* are specified with no *functions,* terminal profile *terminalname* will be modified.

Syntax

CONFIG TERMINAL *terminalname* [*functions*] [*options*]

Functions

/DEL

Deletes the terminal profile named *terminalname.* This *function* requires user confirmation before the profile is deleted unless the /F option is used.

/NEW

Creates a new terminal profile named *terminalname.* Any of the options below can be used with this function.

/Q

Queries the *terminalname* terminal profile and directs the output to STDOUT.

/REN:*newterminalname*
Renames the *terminalname* terminal profile to *newterminalname.*

Options

/F
Used with the /DEL function to force the terminal profile delete without requiring user confirmation.

/AUTOLOGIN:*loginname*
Redefines the autologin user to *loginname.* A *password* can also be specified the same way it would be on the command line for the PASSWORD command (PASSWORD:*password*). If a *password* is specified, surround the *loginname* and *password* with double quotes ("). If no autologin user is specified, the terminal starts with the Login: prompt.

/SUBSYS:*subsystemname*
Redefines the TP subsystem name that the terminal is connected to *subsystemname.* When a new TP subsystem name is specified, a new device name must also be specified. The subsystem name is the name of the file with the subsystem parameters.

/SUBSYSDLL:*filename*
Redefines the TP subsystem DLL filename to *filename.*

/DEVICE:*devicename*
Redefines the device name to *devicename.*

/COMMENT:*string*
Changes the comment to *string.* Strings that contain spaces must be surrounded with double quotes (").

/TERMTYPE:*terminaltype*

Redefines the terminal type that is connected to *terminaltype*. This option forces terminal type settings to the default values defined for terminal *terminaltype*. The *terminaltype* is the name of the file that contains the terminal type settings.

/CONNECT+:*conn1,conn2,...*

Specifies to add the list of connect settings to terminal profile *terminalname*. Valid connect settings are:

CTS	Connect on CTS
DSR	Connect on DSR
RING	Connect on ring indicator
DCD	Connect on DCD
CHAR	Connect on first character
BRK	Disconnect on break
*	All

/CONNECT-:*conn1,conn2,...*

Specifies to remove the list of connect settings from terminal profile *terminalname*. Valid connect settings are defined above.

/FLOW+:*flow1,flow2,...*

Specifies to add the list of flow settings to terminal profile *terminalname*. Valid flow settings are:

XON	XON/XOFF enable
DUP	XON/XOFF full duplex
RTS	RTS/CTS enable
RTSH	RTS input handshaking
DTR	DTR/DSR enable
DTRH	DTR input handshaking

CTSH	CTS output handshaking
DSRH	DSR output handshaking
DCDH	DCD output handshaking
DSRS	DSR input sensitivity
XPC	XPC protocol (Used with XON)
*	All

/FLOW-:*flow1,flow2,...*

Specifies to remove the list of flow settings from terminal profile *terminalname*. Valid flow settings are defined above.

/PARITY:*value*

Redefines the parity setting of the terminal to *value*. Valid parity settings are 0 (no parity), 1 (odd), and 2 (even).

/BAUD:*value*

Redefines the baud rate to *value*. Valid baud rates are 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, 19200, and 38400.

/STOP:*value*

Redefines the number of stop bits to *value*. Valid stop bit values are 1 and 2.

/DATA:*value*

Redefines the number of data bits to *value*. Valid data bit values are 7 and 8.

/INITSTR:*string*

Redefines the modem initialization string to *string*.

/TERMDLL:*filename*

Redefines the terminal type DLL filename to *filename*.

/TCD:*filename*

Redefines the terminal capability data filename to *filename.*

/BUFSIZE:*value*

Redefines the receive buffer size in bytes to *value.* The receive buffer size must be a *value* from 512 to 4096.

/INITSTATE:*value*

Redefines the initial terminal state to *value.* Valid initial terminal state values are 0 (disabled) and 1 (enabled).

/HKLOGIN:*hotkeyString*

Redefines the hotkey used to switch between user logins to *hotkeyString.*

/HKSESS:*hotkeyString*

Redefines the hotkey used to switch between sessions within the same user login to *hotkeyString.*

/HKDIRECT:*hotkeyString*

Redefines the hotkey used to go directly to the session that registers for the direct hotkey to *hotkeyString.* The program that registers for the direct hotkey is typically the Program Selector (*PSEL.EXE*).

/HKCREATE:*hotkeyString*

Redefines the hotkey used to create a new session within the current login to *hotkeyString.*

/HKPRINT:*hotkeyString*

Redefines the hotkey used to do a print screen to *hotkeyString.*

/HKPRINTTOGGLE:*hotkeyString*
Redefines the hotkey used to toggle print screen on and off to *hotkeyString.*

/HKRESET:*hotkeyString*
Redefines the hotkey used to reset the system to *hotkeyString.*

/HKCTRLBRK:*hotkeyString*
Redefines the hotkey used as the **CTRL+BREAK** key sequence to *hotkeyString.*

/HKCTRLC:*hotkeyString*
Redefines the hotkey used as the **CTRL+C** key sequence to *hotkeyString.*

/LOGINCLASS:*classlevel*
Redefines the highest class level allowed to login to the terminal to *classlevel.*

/? (help)
Displays the syntax for the utility and information about the utility's options.

Hotkey Strings

This section describes the *hotkeyString* input for the appropriate CONFIG TERMINAL and CONFIG SYSTEM *options.* A hotkey can consist of up to two shift keys and one non-shift key. They are defined by surrounding each of the keys in braces ({}). The lists below show how to denote shift keys and special non-shift keys.

SHIFT KEYS:

{alt}	-	Either Alt key	{rtalt}	-	Right Alt key
{caps}	-	Caps Lock key	{rtctrl}	-	Right Ctrl key
{ctrl}	-	Either Ctrl key	{rtshift}	-	Right shift key
{ltalt}	-	Left Alt key	{scroll}	-	Scroll Lock key
{ltctrl}	-	Left Ctrl key	{shift}	-	Either shift key
{ltshift}	-	Left shift key	{sysreq}	-	Sys Request key
{num}	-	Num Lock key			

SPECIAL NON-SHIFT KEYS:

{backspace}	-	Backspace key	{minus}	-	(-) on keypad
{break}	-	Break key	{pagedown}	-	Page Down key
{delete}	-	Delete key	{pageup}	-	Page Up key
{down}	-	Down arrow key	{pause}	-	Pause key
{end}	-	End key	{plus}	-	(+) on keypad
{enter}	-	Enter key	{prtscrn}	-	Print screen key
{esc}	-	ESC key	{right}	-	Right arrow key
{f1}	-	F1 key, etc.	{star}	-	(*) on keypad
{home}	-	Home key	{tab}	-	Tab key
{insert}	-	Insert key	{up}	-	Up arrow key
{left}	-	Left arrow key			

Examples:

To define a hotkey as **CTRL+A**, enter the following:

```
{ctrl}{a}
```

To define a hotkey as **ALT+F1**, enter the following:

```
{alt}{f1}
```

The default terminal reset hotkey (**CTRL+ALT+DEL**) would be defined as:

```
{ctrl}{alt}{delete}
```

NOTE: Some of the keys on the top row of the keyboard are not valid with the {ctrl} key.

Security Class Restrictions

Administrator class has full access to all the CONFIG functions. Operator class can view all configurations but cannot modify any. User and Guest classes do not have access to terminal profiles.

CONFIG USER

Description

These are the command line *functions* and *options* for the CONFIG USER utility. The CONFIG USER utility is typically run by going to the full screen (no *functions* or *options* specified). If *options* are specified with no *functions*, user profile *username* will be modified. If a *groupname* is not specified, the default name for the *groupname* will be used.

Syntax

CONFIG USER *username*[*.groupname*] [*function*] [*options*]

Functions

/DEL
: Deletes the user profile named *username*. This *function* requires user confirmation before the profile is deleted unless the /F option is used.

/NEW
: Creates a new user profile named *username* and adds the user as a member to *groupname*. Any of the options below can be used with this option.

/Q

Queries the *username* user profile and directs the output to STDOUT.

/REN:*newUsername*

Renames the *username* user profile to *newUsername.*

Options

/F

Used with the /DEL function to force the user or group membership delete without requiring user confirmation.

/NAME:*string*

Changes the user's formal name to *string.*

/COMMENT:*string*

Changes the comment to *string.*

/USERCOMMENT:*string*

Changes the user comment to *string.*

/ACCOUNTEXPIRES:*numdays*

Sets the account expiration date to *numdays* after today's date. A value of zero (0) indicates the account does not expire.

/PASSWORDEXPIRES:*numdays*

Sets the password expiration date to *numdays* after today's date. A value of zero (0) indicates the password does not expire.

/DEFAULTGROUP:*groupname*

Changes the user's default group to *groupname.* The default group is used when the *username* is specified without a *groupname.* The user must currently be a member of group *groupname.*

/HOMEDRIVE:*drive*

Specifies the *drive* for the user's home directory. This option is only valid when a new user is created. If the /HOMEDRIVE option is not used, the user's home directory is created on drive C.

/ENABLE:*number*

Changes the user state to enabled or disabled, depending on the value of *number*. Valid user state *number*s are zero (0) for disabled and one (1) for enabled.

/AUTODISC:*number*

Changes the login to either auto disconnect when a terminal connection is broken or the not auto disconnect. A login that auto disconnects is reconnected the next time the user logs in. Valid auto disconnect states are zero (0) (do not auto disconnect) and one (1) (auto disconnect).

/SAMETERM:*number*

Changes the requirement of the same terminal in order to reconnect a disconnected login depending on the value of *number*. Valid same terminal values are zero (0) (same terminal is not required to reconnect a login) and one (1) (same terminal is required to reconnect to login).

/WORKDIR:*directory*

Overrides the user's working directory to be *directory*. The working directory is used to override the users home directory as the default current directory when the user logs in to the system.

/G+:*accessletters*

Specifies to add this list of access types (*accessletters*) to those being audited when access is granted. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/G-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is granted. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/D+:*accessletters*

Specifies to add this list of access types (*accessletters*) to those being audited when access is denied. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/D-:*accessletters*

Specifies to remove this list of access types (*accessletters*) from those being audited when access is denied. The *accessletter* is equal to the *permissions* letters described in CONFIG ACCESS.

/USERPROGRAM:*programName*

Overrides the first user program to *programName.* The first user program is the one that gets executed first when the user successfully logs in to the system.

/MAXLOGINS:*number*

Overrides the maximum logins allowed for the user to *number.*

/MAXSESSIONS:*number*

Overrides the maximum sessions allowed for the user in any login to *number.*

/MAXFILES:*number*

Overrides the maximum file handles allowed for the user to *number.*

/MINFILES:*number*

Overrides the minimum file handles that must be available before the user can login to *number.*

/MAXMEMORY:*number*

Overrides the maximum amount of virtual memory (in KB) allowed for the user to *number.*

/MINMEMORY:*number*

Overrides the minimum amount of virtual memory (in Kb) that must be available before the user can login to *number.*

/MAXSEMAPHORES:*number*

Overrides the maximum number of system semaphores that you the user have to *number.*

/MINSEMAPHORES:*number*

Overrides the minimum number of system semaphores that must be available before the user can login to *number.*

/MAXTHREADS:*number*

Overrides the maximum number of threads that the user can have to *number.*

/MINTHREADS:*number*

Overrides the minimum number of threads that must be available before the user can login to *number.*

/LOGIN*day*:*starttime,endtime*

Changes the login restriction of the day of the week to be from *starttime* to *endtime*. The login restriction is used to restrict when a user can be logged into the system. Valid times range from 00:00 to 24:00. Valid *days* are (SUN, MON,TUE,WED,THU,FRI,SAT). If the *starttime* is greater than the *endtime*, the time span will cross midnight (24:00) and carry over into the next day. For example, if the option "/LOGINMON:17:00,6:00" is used, the user can login the system from 5:00pm on Monday through 6:00am on Tuesday.

/? (help)
Displays the syntax for the utility and information about the utility's options.

Security Class Restrictions

Administrator class has full access to all the CONFIG functions. Operator class can view all configurations but cannot modify any. User and Guest classes can view only their User profiles and related Group defaults; they cannot modify anything.

APPENDIX B

ANSI ESCAPE SEQUENCES

INTRODUCTION

This appendix lists all of the escape sequences that you can use with the ANSI command in a full-screen session command interpreter, CMD.

ANSI escape sequences affect cursor positioning, erase functions, and screen graphics.

You must type the escape sequences exactly as shown, without spaces. Each escape sequence begins with the escape character (ASCII code 27) and the left bracket character ([ ). (You cannot press the **ESC** key to produce the escape character.)

CURSOR FUNCTIONS

The following functions affect the movement of the cursor.

Cursor Position

esc[row;colH

or

esc[row;colf

These escape sequences move the cursor to the position specified by the row and column arguments. When no arguments are provided, the cursor moves to the home position (the upper-left corner of the screen).

Cursor Up

esc[nA

This escape sequence moves the cursor up *n* rows without changing columns. If the cursor is already on the top line, no action is taken.

Cursor Down

esc[nB

This escape sequence moves the cursor down *n* rows without changing columns. If the cursor is already on the bottom row, no action is taken.

Cursor Forward

esc[nC

This escape sequence moves the cursor forward *n* columns without changing lines. If the cursor is already in the far-right column, no action is taken.

Cursor Backward

esc[nD

This escape sequence moves the cursor back *n* columns without changing lines. If the cursor is already in the far-left column, no action is taken.

Save Cursor Position

esc[s

This escape sequence saves the current cursor position; the cursor can be returned to the saved position by using the Restore Cursor Position escape sequence.

Restore Cursor Position

esc[u

This escape sequence restores the cursor to the position saved by the last Save Cursor Position escape sequence.

ERASE FUNCTIONS

The following functions erase the screen.

Erase Display

esc[2J

This escape sequence erases the contents of the screen and moves the cursor to the home position (the upper-left corner of the screen).

Erase Line

esc[K

This escape sequence erases from the cursor to the end of the line (including the cursor position).

SCREEN GRAPHICS FUNCTIONS

The following functions affect screen graphics.

Set Graphics Rendition

esc[*g*; ... ;gm

This escape sequence calls the graphics functions specified by the numeric values described in the following list. The specified functions remain in effect until the next occurrence of this escape sequence. This escape sequence works only if the screen device supports graphics.

The *g* variable may have any of the following values:

Value	Function
0	All attributes off
1	Bold on
2	Faint on
3	Italic on
5	Blink on
6	Rapid-blink on
7	Reverse video on
8	Concealed on
30	Black foreground

Value	Function
31	Red foreground
32	Green foreground
33	Yellow foreground
34	Blue foreground
35	Magenta foreground
36	Cyan foreground
37	White foreground
40	Black background
41	Red background
42	Green background
43	Yellow background
44	Blue background
45	Magenta background
46	Cyan background
47	White background
48	Subscript
49	Superscript

The values 30 through 47 meet the ISO 6429 standard.

Set Mode

esc[=sh

This escape sequence changes some of the capabilities and features of the screen. The *s* variable can have any of the following values:

Value	Function
0	40 * 25 black and white
1	40 * 25 color
2	80 * 25 black and white
3	80 * 25 color
4	320 * 200 color
5	320 * 200 black and white
6	640 * 200 black and white
7	Wraps at the end of each line

Reset Mode

esc[=sl

The values for this escape sequence are the same as for Set Mode except that the value 7 turns off line wrapping.

KEYBOARD KEY REASSIGNMENT

esc[#; "*string*"p

where # is the decimal representation of the ASCII character to be mapped. "*string*" is the ASCII string sent to the program when the ASCII key associated with the ASCII character is pressed.

NOTE: Any number of keys can be remapped within the same escape sequence. For example:

esc [3; "key 3"; 4; "key4" p

will remap two keys.

esc [0; #; "*string*" p

where # is the extended ASCII code redefinition. This is typically used to remap function keys.

NOTE: Any number of sequences may be included in front of the "p".

esc ["*string*" p

where the first character of "*string*" defines the ASCII character to be remapped. The remaining characters in "*string*" define the replacement string.

DEVICE STATUS REPORT

```
esc[6n
```

This escape sequence causes the current cursor position (row and column) to be reported through STDIN.

INDEX

Citrix Systems, Inc.
210 University Drive, Suite 700, Coral Springs, FL 33071: Phone (305) 755-0559